(NASA-CR-143936) MISSION ANALYSIS PROGRAM FOR SOLAR ELECTRIC PROPULSION (MAPSEP). VOLUME 3: PROGRAM MANUAL (Martin Marietta Corp.) 584 p HC $13.25 CSCL 21C

N75-31213

Unclas
G3/20 35801

MISSION ANALYSIS PROGRAM FOR SOLAR ELECTRIC PROPULSION (MAPSEP)

CONTRACT NAS8-29666

(Revised)
October, 1974

VOLUME III - PROGRAM MANUAL

Prepared by:

K.R. Huling
R.J. Boain
T. Wilson
P.E. Hong
G.L. Shults

Planetary Systems Mission Analysis and Operations Section
Denver Division
Martin Marietta Corporation

For

National Aeronautics and Space Administration
Marshall Space Flight Center
Huntsville, Alabama

FOREWORD

MAPSEP (Mission Analysis Program for Solar Electric Propulsion) is a computer program developed by Martin Marietta Aerospace, Denver Division, for the NASA Marshall Space Flight Center under Contract NAS8-29666. MAPSEP contains the basic modes: TOPSEP (trajectory generation), GODSEP (linear error analysis) and SIMSEP (simulation). These modes and their various options give the user sufficient flexibility to analyze any low thrust mission with respect to trajectory performance, guidance and navigation, and to provide meaningful system related requirements for the purpose of vehicle design.

This volume is the third of three and contains a description of the internal structure of MAPSEP including logical flow. Prior volumes relate to analytical program description and to operational usage.

TABLE OF CONTENTS

Page

Page

1.0 INTRODUCTION

MAPSEP (Mission Analysis Program for Solar Electric Propulsion) is intended to provide sufficient flexibility to analyze a variety of problems related to trajectory performance, guidance and navigation. However, since low thrust technology is never static, future changes are expected to the models and algorithms contained in MAPSEP. This volume, along with the program listings, is intended to provide the programmer/analyst with sufficient information about MAPSEP structure to enable him to make suitable modifications. The program itself is structured such that computational modules are as self-contained as possible thus facilitating their replacement. It is highly recommended that the programmer/analyst review the two preceding volumes (analytical and user's manuals) before making program changes in order to understand the reasoning behind many of the models and analysis techniques that are coded.

2.0 MACROLOGIC

MAPSEP is composed of three primary modes: TOPSEP, GODSEP and SIMSEP (Figure 2-1). A fourth primary mode, REFSEP, is actually a submode of TOPSEP in a functional sense. In addition, a secondary mode, TRAJ, is used by all four primary modes to provide integrated trajectory information. As described in both the Analytic and User's Manuals, the primary modes each serve a specific function in the mission and system design sequence.

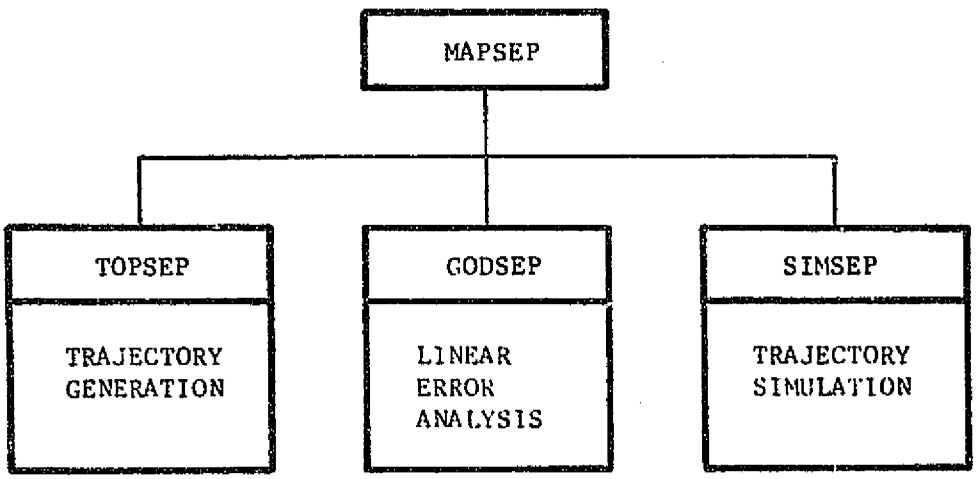


Figure 2-1. MAPSEP Modes

All of the routines and structure of MAPSEP are constructed to minimize core storage (thus reducing turn-around time and computer run cost) yet retain the flexibility needed for broad analysis requirements. Furthermore, routines are built as modular as possible to reduce the difficulties in future modifications and extensions.

2.1 Input/Output

The user interface or input to MAPSEP is primarily through cards using the NAMELIST feature, with supplementary means depending upon mode and function (Table 2-1). All modes require the $TRAJ namelist which defines the nominal trajectory and subsequent

	INPUT			OUTPUT	
Mode	Namelist	Formated Cards	Tape (or disc)	Punched Cards	Tape (or disc)
TOPSEP	$TRAJ $TØPSEP	None	STM	None	STM GAIN
GODSEP	$TRAJ $GØDSEP $GEVENT	Event Data	STM GAIN	States Covariances Guidance	STM GAIN SUMARY
SIMSEP	$TRAJ $SIMSEP $GUID	None	STM	Statistics	STM GAIN SUMARY
REFSEP	$TRAJ	Print Events	STM	None	STM

TABLE 2-1. MAPSEP User Input/Output

mode usage. However, if recycling or case stacking is performed it is not necessary to input $TRAJ again unless desired. The second namelist required for each mode corresponds to mode peculiar input and bears the name of that particular mode. Additional namelist, formated cards, and tape input are generally optional. Besides the standard printout associated with MAPSEP, auxiliary output can be obtained which will facilitate subsequent runs.

From an operational viewpoint, MAPSEP employs a maximum of six data files (Table 2-2). Most of these files are not normally saved from run to run, the primary exceptions being STMFILE and GAINFIL used in GODSEP.

I/O File Number	File	Mode Usage		
		TOPSEP AND REFSEP	GODSEP	SIMSEP
TAPE 3	STM	$TRAJ namelist	$TRAJ namelist, trajectory and state transition matrix data	$TRAJ namelist
TAPE 4	GAIN	-	a-priori covariances and filter gain matrices	$GUID namelists
TAPE 5	INPUT	input data	input data	input data
TAPE 6	OUTPUT	printout	printout	printout
TAPE 7	PUNCH	-	punched covariances	punched statistics
TAPE 8	SUMMARY	trajectory summaries	event data summaries	$SIMSEP namelist

TABLE 2-2. Data Files

2.2 Overlay Structure

The structure of MAPSEP is organized into three levels of "overlays" which are designed to minimize total computer storage. At any given time, only those routines which are in active use are loaded into the working core of the computer. The main overlay (Figure 2-2) is always in core and contains the main executive, MAPSEP, and all utility routines that are common to the three modes. The primary overlays contain key operating routines of each mode, that is, those routines which are always needed when that particular mode is in use. Also included as a primary overlay is the data initialization routine, DATAM, where $TRAJ namelist is read, trajectory and preliminary mode parameters are initialized, and appropriate parameters are printed out.

The secondary overlays contain routines which perform various computations during a particular operational sequence. Included are data initialization routines, analgous to DATAM, which operate on mode peculiar input and perform mode initialization. An example of core usage in the changing overlay structure may be provided by a standard error analysis event sequence. Error analysis initialization is performed by the overlay DATAG. Transition matrices are then read from the STM file, the state covariance is propagated to a measurement event, and the overlay MEAS is called, which physically replaces, or overlays, the same core used previously by DATAG. Similarly at a guidance event, overlay TRAJ will replace MEAS to compute target sensitivity matrices and overlay GUID will then replace TRAJ to compute guidance corrections. Overlay switching is performed internally and is transparent to the user.

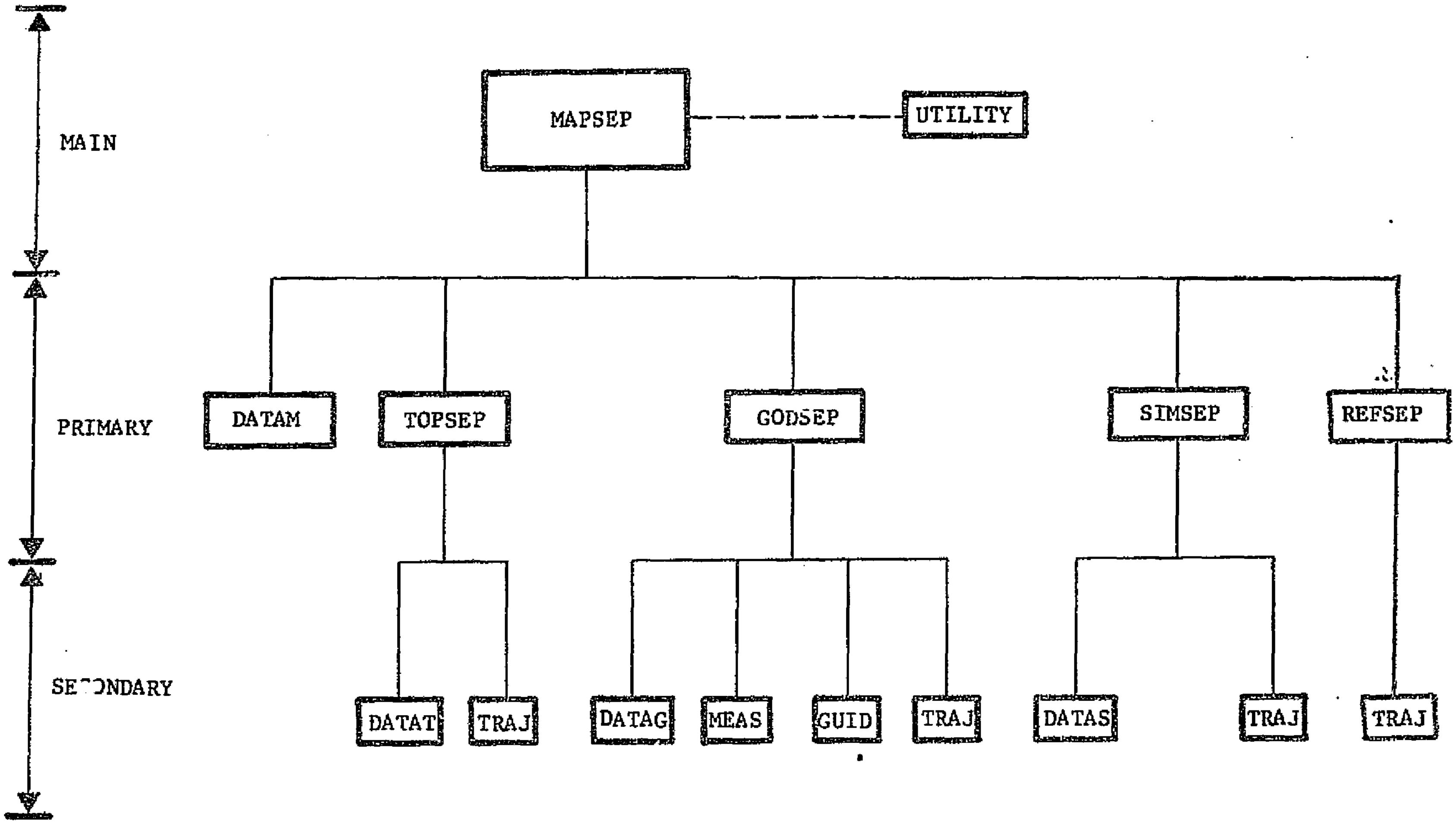


Figure 2-2. OVERLAY STRUCTURE

2.3 Subroutine Hierarchy

Each major overlay is supported by a number of routines, some of which are contained in that overlay, others are in higher overlays. Figures 2-3, 2-4, 2-5, 2-6, and 2-7 illustrate the subroutine hierarchy for the major overlays TRAJ, TOPSEP, GODSEP, SIMSEP, and REFSEP, respectively. Multiple calls to subroutines and entry points are not shown, but may be found in the detailed subroutine descriptions (Chaper 3). The hierarchies also do not distinguish between routines called from different overlays.

2.4 Blank Common

One convenient feature of the CDC 6000 series computer (on which MAPSEP was developed), is the ability to specify the location in core where blank common is loaded. This allows blank common to be loaded behind the longest secondary overlay to be loaded for the current mode. Thus, the length of blank common may be adjusted merely by changing the amount of core requested for the job. The resultant convenience factor is a core saving on many runs. Wherever possible, large arrays whose dimensions vary as a function of input parameters are loaded in blank common. Each mode in its data overlay computes the locations of these arrays as required by the input. Each mode starts using blank common from the first word, and defines for the TRAJ overlay the first available word of blank common it may access. TRAJ stores all information evaluated for integration steps in blank common. For an example of the disparity in blank common lengths required for different runs, the sample error

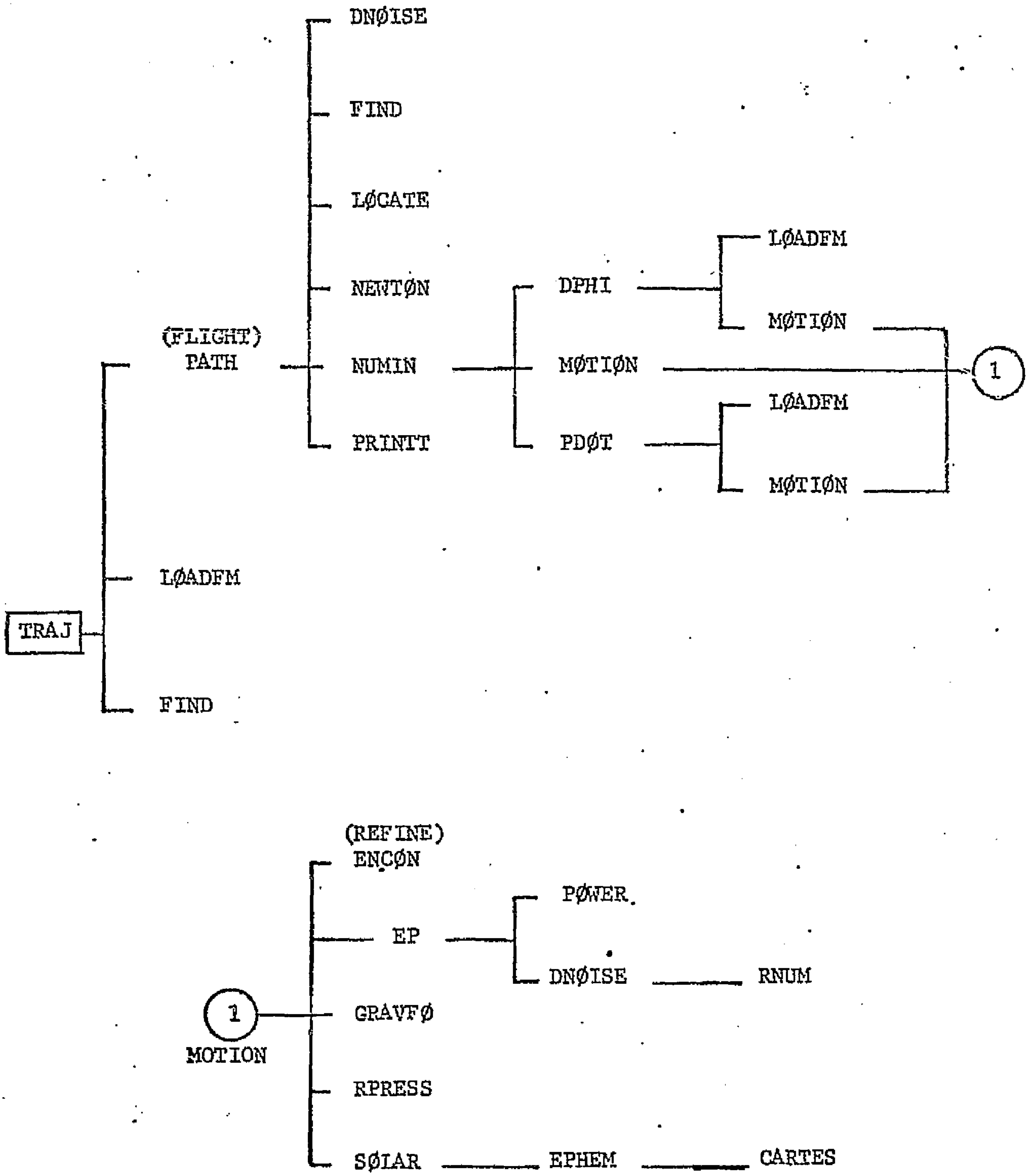


Figure 2-3. TRAJ Subroutine Hierarchy

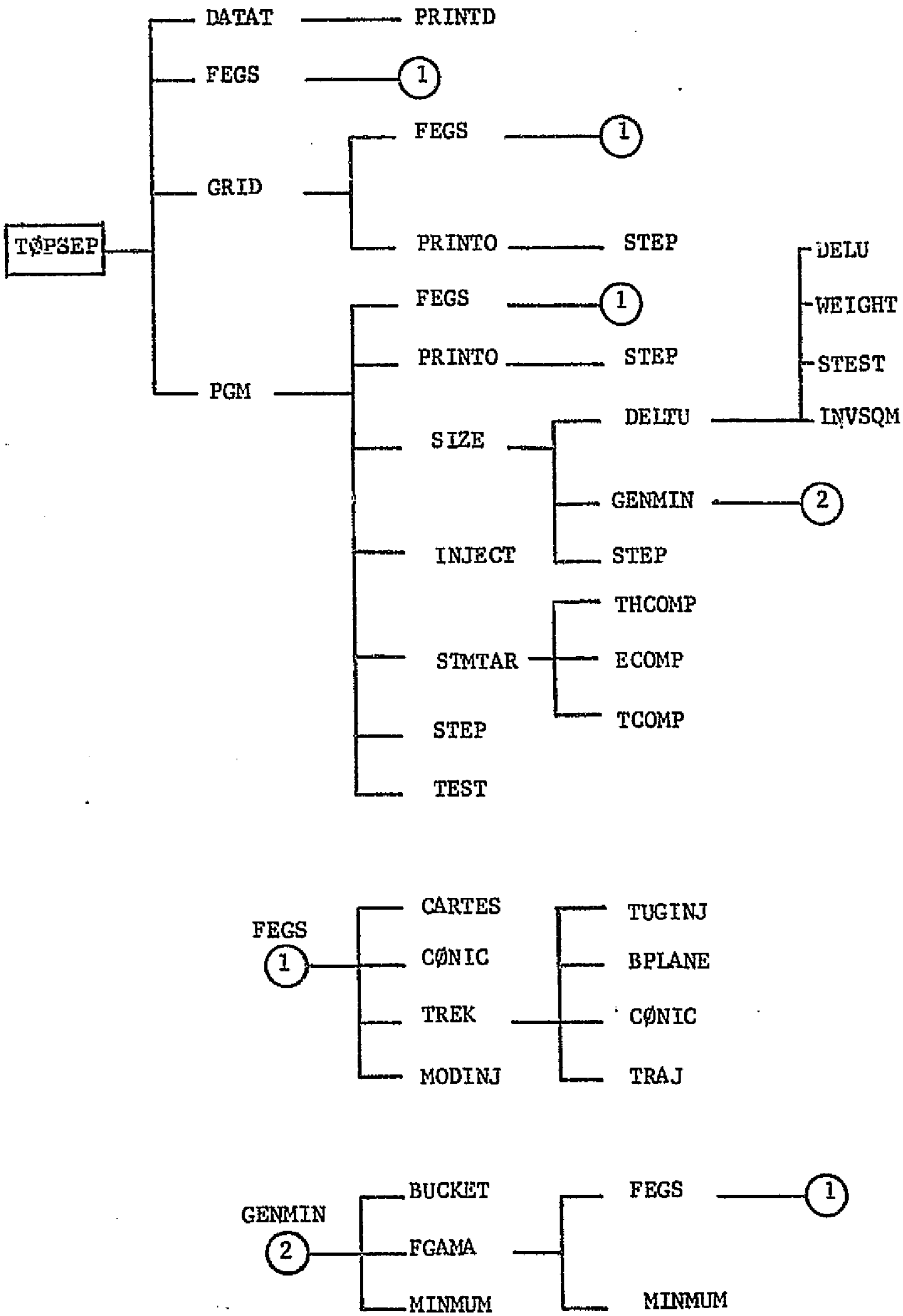


Figure 2-4. TOPSEP Subroutine Hierarchy

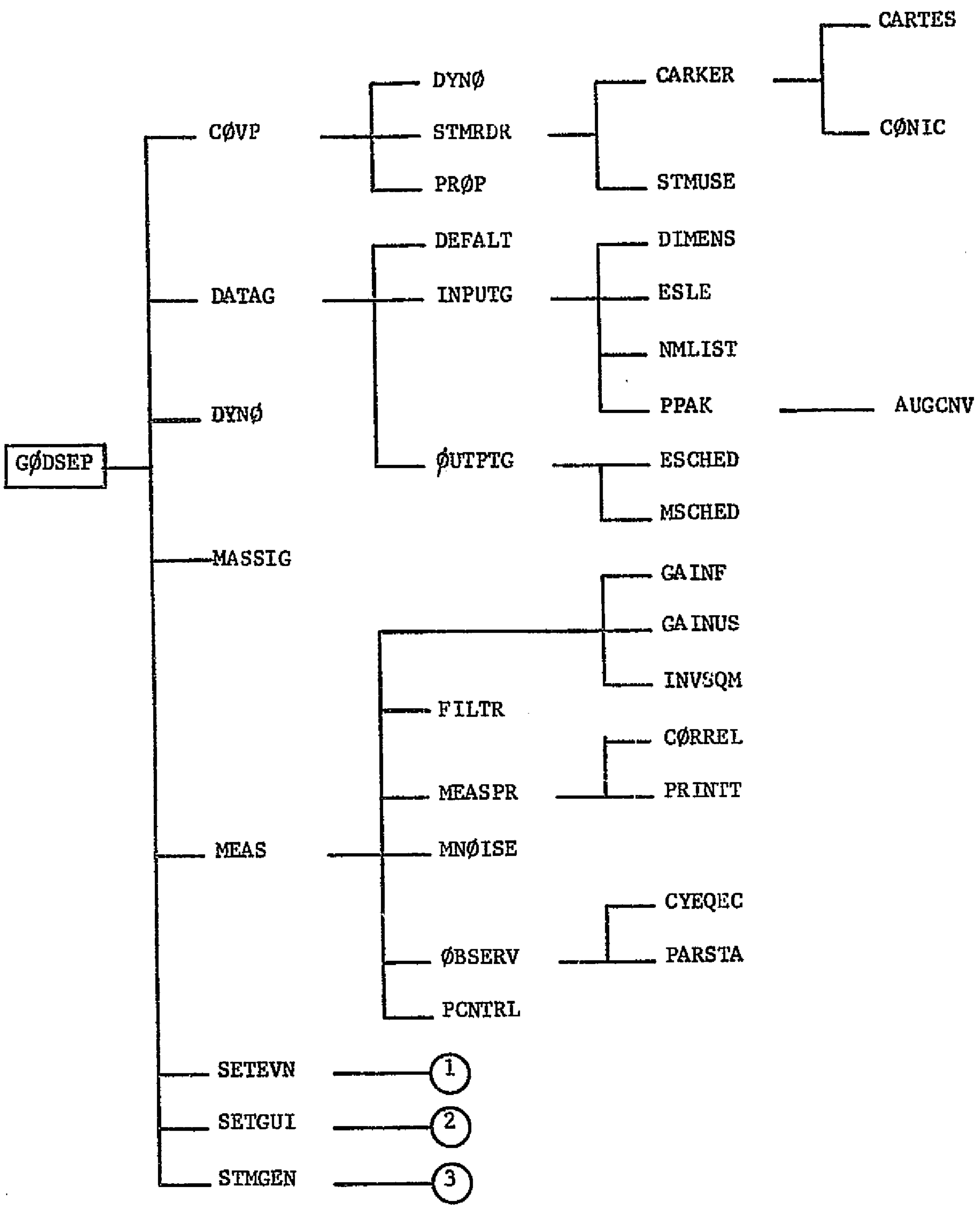


Figure 2-5. GØDSEP Subroutine Hierarchy

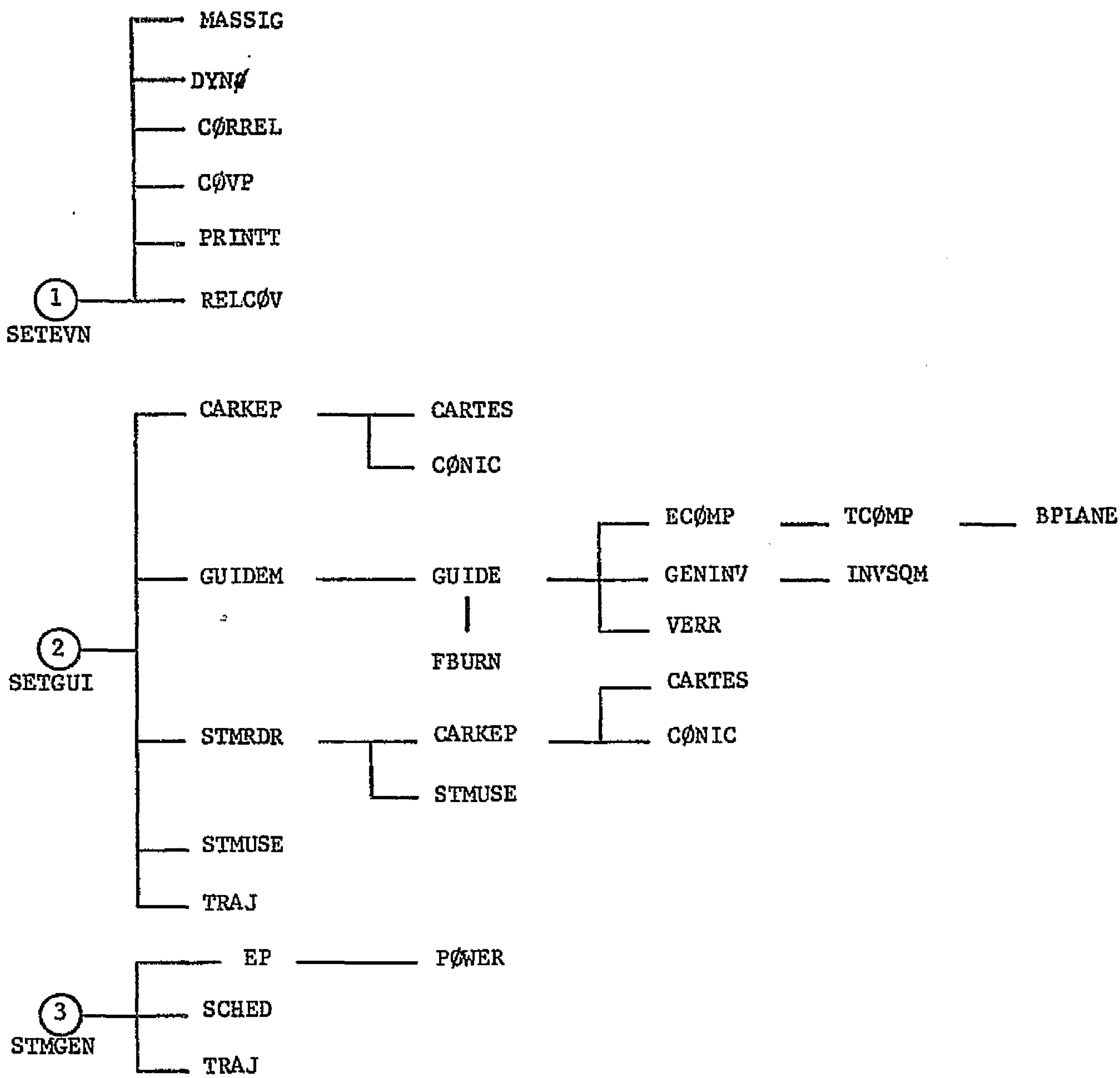


Figure 2-5. GØDSEP Subroutine Hierarchy (Continued)

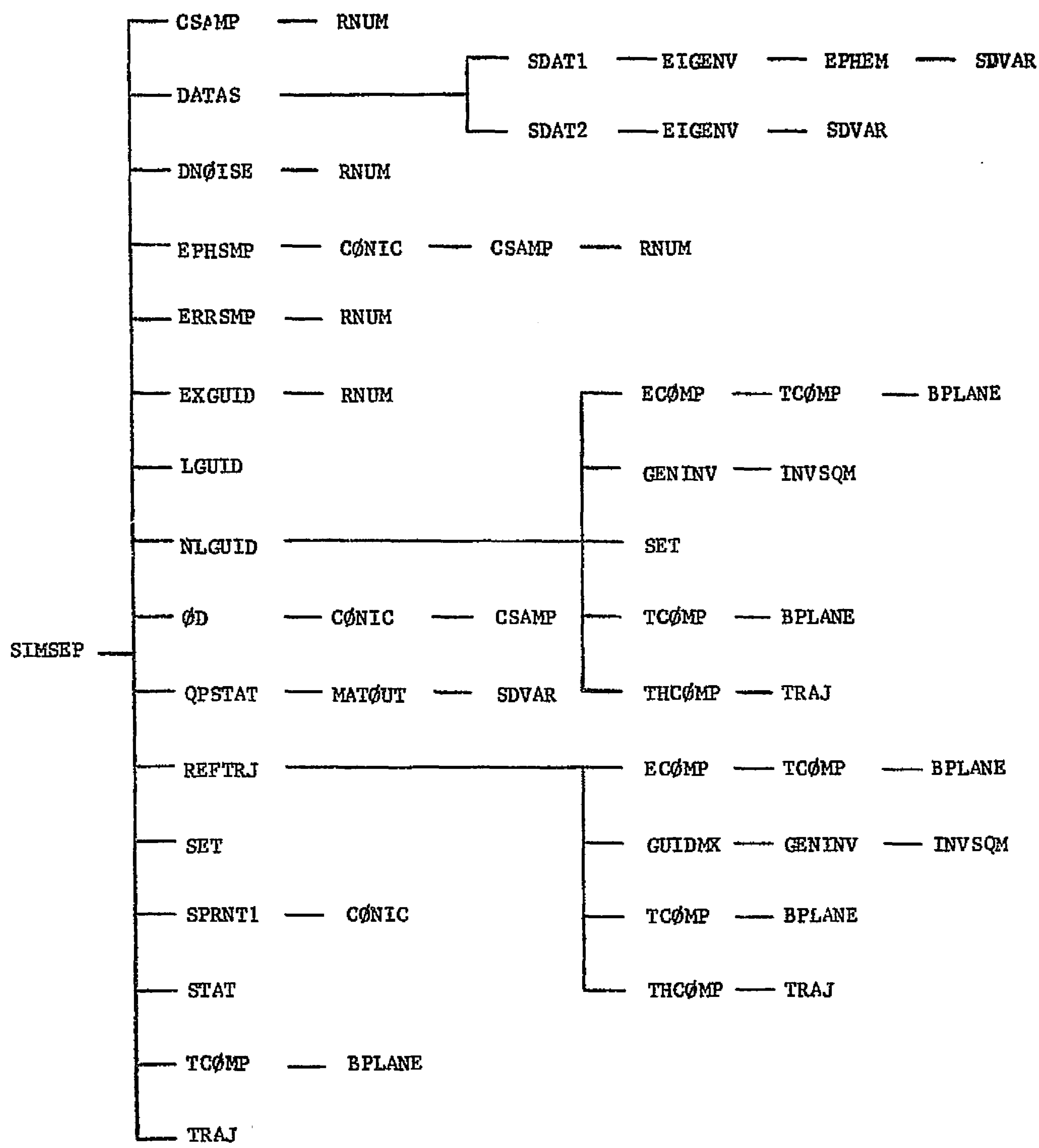


Figure 2.6 SIMSEP Subroutine Hierarchy

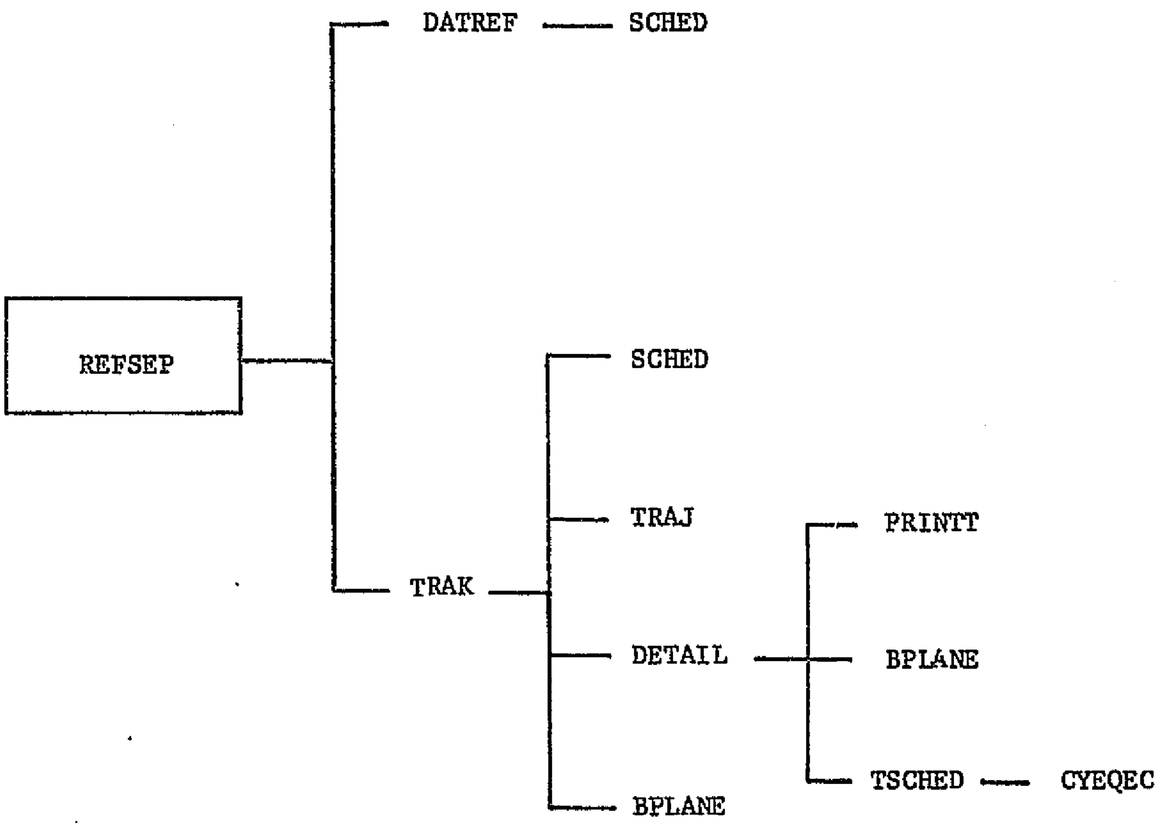


Figure 2-7. REFSEP Subroutine Hierarchy

analysis included in the User's Manual (Vol. II Sec. 3.2.2) requires 5184 decimal or 12100 octal words of blank common. The same run without guidance would require only 2304_{10} (4400_8) words of blank common. A TØPSEP run which does no targeting or optimization -- merely integrates a reference trajectory -- requires less than 100_{10} words of blank common.

2.5 Program Loading

The recommended usage of MAPSEP, which also minimizes computer core for a given run, is to load only those overlays and related routines which are necessary for the run. This is performed by "satisfying" from a master library file which contains all of the MAPSEP routines. In this case the deck necessary to run MAPSEP consists only of the overlay structure and the input data decks. The advantage is a direct result of not having to load all utility routines in the main overlay. Instead, the utility routines are loaded only in the overlays where they are used. In addition, blank common can easily be set to the size necessary to handle specific mode runs, thus, reducing further the overall core requirements. Figure 2-8 illustrates core utilization when satisfying from a library file.

If a library file is not used, then the utility routines would be loaded after the I/O buffers in Figure 2-8 and before the primary overlays. Although the core required for each primary overlay would be smaller, the total core (utility + primary) would be greater. Furthermore, blank common would start at the end of the last routine

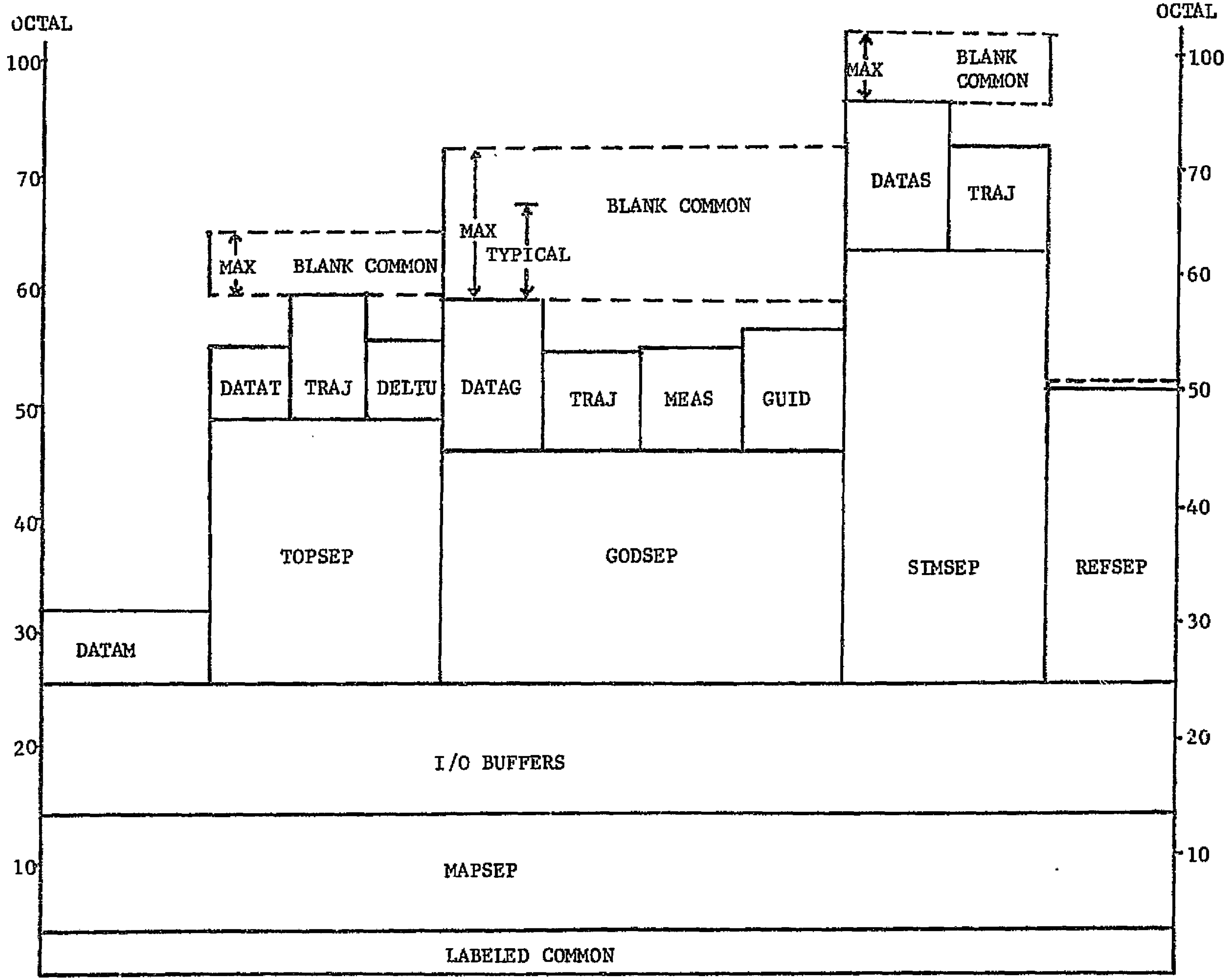


Figure 2-8. Core Utilization (with library file)

(DATAS) so that the overall core penalty, if the entire program is loaded at once, would be approximately 3k to 20k, depending upon the operating mode.

For those users who can vary the amount of blank common storage in their runs, a guideline to estimate the total MAPSEP core requirements is given below. Blank common length is related directly to the dimension of the dynamic state (NDIM) used in transition matrix (STM) computation, and, the total augmented (knowledge) state (NAUG). The values of "program" and "blank common" must be added to compute the total decimal core for a CDC 6500. Other operating systems must scale these requirements appropriately.

TOPSEP:	program = 23400	(N = number of control parameters)
	blank common = $800 + 68(N)+(N)^2$	
GODSEP:	program = 23900	
	blank common = $100 + 9\ (NDIM)^2$	(if STM created)
	$= 100 + 9\ (NDIM)^2 + 5\ (NAUG)^2$	(if STM used)
	$= 100 + 13\ (NAUG)^2$	(if PDOT used)
SIMSEP:	program = 39100	(N = number of guidance events)
	blank common = $900 + N(NAUG)^2$	
REFSEP:	program + blank common = 21000	

2.6 Labeled Commons

The labeled common blocks are grouped according to the principal overlays in which they are used: MAPSEP, TOPSEP, GODSEP, and SIMSEP. The type of each variable will be specified as follows:

Type	Designation
Real	R
Integer	I
Logical	L
Hollerith	H
Assigned GØ TØ Statements	S

All units will be in km, km/sec, days, radians, kg, kW, km/sec^2, or km^3/sec^2 unless otherwise noted.

The following index of common blocks is intended to facilitate their location by the reader.

Common	Principal Overlay	Page
PRINT	TOPSEP	28
PRINTH	TOPSEP	28
PRØPI	GODSEP	46
PRØPR	GODSEP	46
SCHEDI	GODSEP	47
SCHEDR	GODSEP	49
SIMLAB	SIMSEP	52
SIM1	SIMSEP	53-A
SIM2	SIMSEP	53-B
STØREC	SIMSEP	53-C
TARGET	MAPSEP	18-B
TIME	MAPSEP	19
TØP1	TOPSEP	28
TØP2	TOPSEP	32
TRAJ1	MAPSEP	19
TRAJ2	MAPSEP	22
TRKDAT	MAPSEP	26
TUG	TOPSEP	34
WØRK	MAPSEP	26

2.6.1 MAPSEP Labeled Commons

Most common blocks that appear in MAPSEP primarily are used to save information created by the overlays DATAM and TRAJ. Other common blocks that appear in MAPSEP are used to transmit information from the Conic subroutines.

a) Common/CØNST/program constants

Name	Dimension	Type	Definition
AU	1	R	149597893. (km/AU)
BIG	1	R	10^{20}
ECEQ	3 x 3	R	Transformation matrix from Earth equatorial to Earth ecliptic coordinates
FØP	1	R	10^{-15}
FØV	1	R	10^{-25}

Name	Dimension	Type	Definition
GHZERØ	1	R	Greenwich Hour angle at launch
ØMEGAG	1	R	6.300388099 Earth rotation rate in rad/day
PI	1	R	3.14159 . . . . (PI)
RAD	1	R	57.29 . . . . (deg/rad)
SMALL	1	R	10^{-20}
TM	1	R	86400.0 (sec/day)

b) Common/CØNICS/Osculating conic parameters

Name	Dimension	Type	Definition
PV	3	R	Eccentricity unit vector
QV	3	R	Unit vector orthogonal to WV and PV
WV	3	R	Angular momentum unit vector
RM	1	R	Position Magnitude
VM	1	R	Velocity Magnitude
RDV	1	R	$\underline{r} \cdot \underline{v}$
H	1	R	Angular momentum magnitude
P	1	R	Semi-latus rectum

c) Common/EDIT/future modification storage

Name	Dimension	Type	Definition
EDIT	50	R	Miscellaneous storage array; intended for use by temporary modifications until permanent storage (labeled and blank common) is arranged.

Common/ENCØN/local variables for subroutine ENCØN (see PP. 534, 535)

d) Common/EPHEM/ ephemeris constants

Name	Dimension	Type	Definition
CECC	4 x 10	R	Eccentricity constants of the planets

Name	Dimension	Type	Definition
CINC	4 x 10	R	Inclination constants of the planets
CMEAN	4 x 10	R	Mean anomaly constants of the planets
CØMEG	4 x 10	R	Longitude of the ascending node constants of the planets
CØMEGT	4 x 10	R	Longitude of periapsis constants of the planets
CSAX	2 x 10	R	Semi-major axis constants of the planets
DJ1900	1	R	Julian Date of January 0.5, 1900
EMN	15	R	Lunar ephemeris constants
PLANET	11	H	Hollerith label for the planets
PMASS	11	R	Planetary gravitational constants
PRADIS	11	R	Planetary radii
SMASS	1	R	Solar gravitational constant
SPHERE	11	R	Planetary SOIs
SRADIS	1	R	Radius of the sun
SUN	1	H	Hollerith label for the sun

f) Common/IASTM/Sensitivity Matrix Parameters

Name	Dimension	Type	Definition
IASTM	1	I	Flag designating method of computing targeting sensitivity matrix
IJH	2x30	I	Array of flags identifying active controls
LISTAR	6	I	Array of flags identifying active targets
THETA	6x20	R	Sensitivity of final state to changes in thrust controls
PHI	6x6	R	Sensitivity of final state to changes in initial state (STM)

g) Common/TARGET/Osculating Conic Conditions

Name	Dimension	Type	Definition
VCA	1	R	Speed at closest approach.
CA	1	R	Radius of closest approach
TCA	1	R	Time of closest approach
BDT	1	R	$\underline{B} \cdot \underline{T}$
BDR	1	R	$\underline{B} \cdot \underline{R}$
TSI	1	R	Time of sphere of influence crossing
VHP	1	R	Hyperbolic excess velocity
SMA	1	R	Semi-major axis
ECC	1	R	Eccentricity
XINC	1	R	Inclination
ØMEGA	1	R	Longitude of the ascending node
SØMEGA	1	R	Argument of periapsis
XMEAN	1	R	Mean anomaly
TA	1	R	True anomaly
F1	1	R	Hyperbolic anomaly
B	1	R	B-vector magnitude
BV	3	R	B-vector
TAIM	1	R	Theta aim (angle between the B-vector & Taxis)
SV	3	R	S-vector (unit vector in direction of VHP vector)

h) Common/TIME/time parameters

Name	Dimension	Type	Definition
EPØCH	1	R	Julian Date of launch
TCP	1	R	Total CP time required to integrate a trajectory
TDUR	1	R	Trajectory termination time from launch in seconds
TEND	1	R	Trajectory termination time from launch in days
TEVNT	1	R	Trajectory event time in seconds
TRCA	1	R	Time of closest approach
TREF	1	R	Trajectory start time from launch, in seconds
TSØI	1	R	Time at the sphere of influence of the target body
TSTART	1	R	Trajectory start time
TSTØP	1	R	Actual trajectory termination time

i) Common/TRAJ1/trajectory propagation parameters

Name	Dimension	Type	Definition
ACC	1	R	Integration step-size scale factor
ALPHA	1	R	Inverse semi-major axis of the reference conic
APERT	3 x 12	R	Gravitational acceleration vectors due to the perturbing bodies
APRIM	3	R	Gravitational acceleration vector due to the primary body
ATØT	3	R	Total differential acceleration vector
BØDY	3	H	Hollerith label of the planets included in the integration
DRMAX	3	R	Maximum deviation from the reference conic
ENGINE	20	R	Array that defines the thrust and power subsystems

Name	Dimension	Type	Definition
FRCA	1	R	Fraction of the semi-major axis of the target planet to begin closest approach tests
GM11	3	R	Matrix of partial derivatives for transition matrix integration
GM12	3	R	Matrix of partial derivatives for transition matrix integration
GM21	3	R	Matrix of partial derivatives for transition matrix integration
GM22	3	R	Matrix of partial derivatives for transition matrix integration
GT	3 x 3	R	Matrix of partial derivatives for transition matrix integration
GTAU1	3 x 3	R	Diagonal matrix of inverse correlation times (first process)
GTAU2	3 x 3	R	Diagonal matrix of inverse correlation times (second process)
G11	3 x 3	R	Matrix of partial derivatives for transition matrix integration
G12	3 x 3	R	Matrix of partial derivatives for transition matrix integration
G22	3 x 3	R	Matrix of partial derivatives for transition matrix integration
QNØISE	6 x 6	R	Matrix of process noise
RCA	1	R	Local variable used in TRAJ
RPACC	3	R	Acceleration vector due to radiation pressure
RSTØP	1	R	Desired stopping radius
SCMASS	1	R	Initial spacecraft mass
SCMVAR	1	R	Initial spacecraft mass variation
STATEO	8	R	First three elements are the initial position vector

Name	Dimension	Type	Definition
			Second three elements are the initial velocity vector
			Seventh element is the position magnitude
			Eighth element is the velocity magnitude
TCPI	1	R	CP time at the beginning of the integration
THRACC	3	R	Acceleration vector due to thrust
THRUST	10 x 20	R	Array used to define the operation of the thrust subsystem
TNØISE	6	R	First three elements contain thrust noise for the first process
			Second three elements contain thrust noise for the second process
UENC	3	R	Reference conic position vector
UENCM	1	R	Reference conic position magnitude
UP	3 x 12	R	Position vectors of all the bodies included in the integration
UREL	3 x 12	R	Position vectors of the spacecraft relative to all the bodies considered in the integration
URELM	12	R	Magnitudes of UREL
UTRUE	3	R	S/C position vector relative to the primary body
UTRUEM	1	R	S/C position magnitude relative to the primary body
VENC	3	R	Reference conic velocity vector
VENCM	1	R	Reference conic velocity magnitude
VP	3 x 12	R	Velocity vectors of all the bodies considered in the integration
VREL	3 x 12	R	Velocity vectors of the spacecraft relative to all the bodies considered in the integration

Name	Dimension	Type	Definition
VRELM	12	R	Magnitudes of VREL
VTRUE	3	R	S/C velocity vector relative to the primary body
VTRUEM	1	R	S/C velocity magnitude relative to the primary body
WPØWER	1	R	Power available
XPRINT	1	R	Print interval
ZK	3	R	Direction cosines of the reference star

j) Common/TRAJ2/Trajectory Flags

Name	Dimension	Type	Definition
IAUGDC	10	I	Array of flags used to augment the state for transition matrix or covariance integration
ICALL	1	I	Flag used to initialize TRAJ or to initialize TRAJ and to start integration or to continue integration from the previous time
IENRGY	1	I	Flag that determines the kind of power subsystem
IEVENT	1	S	Local variable used in TRAJ
IEVNT1	1	S	Local variable used in TRAJ
IEVNT2	1	S	Local variable used in TRAJ
IEVNT3	1	S	Local variable used in TRAJ
IEP	1	I	Flag used to locate information about the ephemeris body (1 = Sun, 2 = Earth,...)
IMØDE	1	I	Submode designation in TØPSEP
INIT	1	I	MAPSEP initialization flag
INTEG	1	I	Flag used to determine the type of equations to be integrated

Name	Dimension	Type	Definition
INTEG2	1	S	Local variable used by TRAJ
INTEG3	1	S	Local variable used by TRAJ
IPFLAG	1	I	Flag used to designate a control phase change
IPHASE	1	S	Local variable used in TRAJ
IPHASO	1	S	Local variable used in TRAJ
IPHAS1	1	S	Local variable used in TRAJ
IPHAS2	1	S	Local variable used in TRAJ
IPLACE	1	S	Local variable used in TRAJ
IPRI	1	I	Flag used to locate information about the primary body
IPRINT	1	I	Flag used to manipulate the trajectory print options'
IPRT	1	S	Local variable used in TRAJ
IPRT1	1	S	Local variable used in TRAJ
IRECT	1	I	Flag used to control rectification
ISTEP	1	I	Number of integration steps taken
ISTMF	1	I	Flag used to control STM file use
ISTØP	1	I	Flag used to set the trajectory termination logic
ITEST	1	S	Local variable used in TRAJ
ITP	1	I	Flag used to locate information about the target body
ITRAJ	1	I	Local variable used in TRAJ
JPFLAG	1	I	Flag used to designate a primary body change
JPHAS1	1	S	Local variable used in TRAJ
JPHAS2	1	S	Local variable used in TRAJ

Name	Dimension	Type	Definition
JPHAS3	1	S	Local variable used in TRAJ
JTEST	1	S	Local variable used in TRAJ
KSTØP	1	S	Local variable used in TRAJ
KTRAJ	1	I	Flag used to designate whether to test for control phase changes
KUTØFF	1	I	Flag used to designate the actual trajectory stopping criteria
LPRINT	1	S	Local variable used in TRAJ
LØCAL	1	S	Local variable used in TRAJ
LØCDM	1	I	Location of the output mass variation in blank common
LØCDT	1	I	Location of the temporary derivatives in blank common
LØCDY	1	I	Location of the nominal derivatives in blank common
LØCET	1	I	Location of the integration event time in blank common
LØCFI	1	I	Location of the F matrix in blank common
LØCFØ	1	I	Location of the covariance to be integrated in blank common
LØCH	1	I	Location of the integration step-size in blank common
LØCM	1	I	Location of the output mass in blank common
LØCPR	1	I	Location of the integration print time in blank common
LØCPT	1	I	Location of the actual print time in blank common
LØCR	1	I	Location of the stored position magnitudes in blank common
LØCS	1	I	First location in blank common that can be used by TRAJ

Name	Dimension	Type	Definition
LØCT	1	I	Location of the stored trajectory times in blank common
LØCTC	1	I	Location of the output transition matrix or covariance in blank common
LØCTE	1	I	Not used
LØCYC	1	I	Location of the nominal integrated solution in blank common
LØCYP	1	I	Location of the intermediate integrated solution in blank common
LØCYT	1	I	Location of the temporary integrated solution in blank common
LØCX	1	I	Location of the trajectory time in blank common
MEQ	1	I	Total number of equations to be integrated
MEQS	1	I	Dimensions of the augmented transition matrix or covariance
MEQ8	1	I	MEQ minus 8
MEVENT	1	I	Flag used to set event detection logic
MØDE	1	I	Flag used to set the MAPSEP mode of operation (TØPSEP, GØDSEP, SIMSEP)
MPLAN	1	I	Number of bodies included in the integration
MSTØP	1	S	Local variable used in TRAJ
NB	11	I	Planet codes of the bodies to be included in the integration
NBØD	1	I	Number of bodies in NB
NEP	1	I	Planet code of the ephemeris body
NLP	1	I	Planet code of the launch body
NØISED	1	I	Flag used to turn off the noise for the simulation mode

Name	Dimension	Type	Definition
NPHASE	1	I	Flag to test for primary body changes
NPRI	1	I	Planet code of the primary body
NPRINT	1	S	Local Variable used in TRAJ
NRECT	1	I	Number of rectifications executed
NSTØP	1	S	Local Variable used in TRAJ
NTP	1	I	Planet code of the target body
NTPHAS	1	I	Number of the current control phase

k) Common/TRKDAT/ — Tracking Data

Name	Dimension	Type	Definition
ELVMIN*	1	R	Minimum elevation angle for tracking
IOBS*	1	I	Location in STALØC of astronomical observatory
KARDS*	1	I	Number of formatted print schedule cards following the $TRAJ namelist
NSTA*	1	I	Number of S/C tracking stations
STALØC	3x9	R	Station location coordinates
STARDC	3x9	R	Star direction cosines

* Variables exclusive to the REFSEP mode

l) Common/WØRK/ — Working Storage

Name	Dimension	Type	Definition
WØRK	200	R	Array used as local variables to conserve core locations

2.6.2 TOPSEP Common Blocks

a) CØMMØN/CYCLE/TOPSEP Cycle Flag

Name	Dimension	Type	Definition
ICYCLE	1	I	Mode cycle flag. = 0, Do not store namelist variables on disc. = 1, Store namelist variables on disc.

b) CØMMØN/GRID/Blank Common Locations

Name	Dimension	Type	Definition
LØCE1	1	I	Blank common location of the target errors associated with the first step of the control grid.
LØCE2	1	I	Blank common location of the target errors associated with the second step of the control grid.
LØCEM1	1	I	Blank common location of the target error indices associated with the first step of the control grid.
LØCEM2	1	I	Blank common location of the target error indices associated with the second step of the control grid.
LØCEN	1	I	Blank common location of the nominal trajectory target errors in the grid mode.
LØCF1	1	I	Blank common location of the performance indices associated with the first step of the control grid.
LØCF2	1	I	Blank common location of the performance indices associated with the second step of the control grid.

c) COMMON/PRINT/Printout Parameters

Name	Dimension	Type	Definition
CNTRØL	20	R	Initial values of all possible controls other than thrust controls.
ETLØUT	6	R	Target tolerances in print-out units.
GØUT	20	R	Performance gradient in print-out units.
HØUT	10x22	R	Perturbation array in print-out units.
KNTRØL	20	H	Hollerith names of controls in CNTRØL.
SØUT	120	R	Sensitivity matrix in print-out units.
TARØUT	6	R	Desired target values in printout units.

d) CØMMØN/PRINTH/Printout Labels

Name	Dimension	Type	Definition
LABELT	6	H	Hollerith names of chosen targets.
LABEL	25	H	Hollerith names of all possible targets.

e) CØMMØN/TØP1/TOPSEP Parameters - Real Variables

Name	Dimension	Type	Definition
BTØL	1	R	Tolerance on control bounds.
CHI	1	R	In plane Δv direction angle at injection.
CNVRTT	6	R	Conversion constants from input units to internal units for selected targets.

Name	Dimension	Type	Definition
CNVRTU	20	R	Conversion constants from input units to internal units for selected controls.
CTHETA	1	R	Cosine of optimization angle.
DELVO	1	R	Injection ΔV .
DFMAX	1	R	Maximum increase allowed in the cost index (F) per iteration.
DPSI	6	R	Target error to be removed during current iteration.
DP2	1	R	Estimated region of linearity in the control space.
E	6	R	Target errors of the current trajectory.
EMAG	1	R	Target error index.
EPSØN	1	R	Scalar multiple for control perturbations.
ETØL	6	R	Target tolerances.
ETR	6x6	R	Array of target errors of the reference and all trial trajectories evaluated during a single iteration.
F	1	R	Performance index of the current trajectory.
FTR	6	R	Vector of performance indices of the reference and all trial trajectories evaluated during a single iteration.
G	20	R	Performance gradient.
GAMA	1	R	Scale factor providing the best control change.
GAMMA	6	R	Vector of trial trajectory control change scale factors.

Name	Dimensions	Type	Definition
GTRIAL	5	R	One-dimensional search constants.
G	10x22	R	Control perturbation array.
HMULT	20	R	Vector of scalar multiples of the H array to determine the second step of all controls in the control grid.
ØPTEND	1	R	Cosine of the optimization angle which is used to test convergence in the targeting and optimization mode.
ØSCALE	1	R	Scale on the performance index when simultaneously targeting and optimizing.
PCT	1	R	Percentage of the target error to be removed during an iteration.
PRTURB	20	R	Vector of control perturbations; summary of H array.
PSI	1	R	Out of plane ΔV direction angle at injection.
P1	6	R	Vector of net cost values for the reference and all trial trajectories evaluated during a single iteration.
P1P2	6	R	Vector of combined target error indices and net cost values for the reference and all trial trajectories evaluated during a single iteration.
P2	6	R	Vector of target error indices for the reference and all trial trajectories evaluated during a single iteration.
S	6x20	R	Target sensitivity matrix.
STATR	8x6	R	Array of initial states for the reference and all trial trajectories evaluated during a single iteration.

Name	Dimensions	Type	Definition
STØL	1	R	Test variable for determining linearly dependent columns of the weighted sensitivity matrix.
TARGET	6	R	Vector of desired target values.
TARNØM	6	R	Target values evaluated for the reference trajectory.
TARPAR	6	R	Target values of the most recently generated trajectory.
TARTØL	25	R	Vector of all possible target tolerances.
TARTR	6x6	R	Target values of the reference and all trial trajectories evaluated during a single iteration.
TLØW	1	R	Limit of target error index below which optimization only is performed.
TUP	1	R	Limit of target error index above which simultaneous targeting and optimization is discontinued and targeting only is initiated.
U	20	R	Selection of controls for the specified mode run.
UWATE	20	R	User input weights on controls.
VPARK	1	R	Circular parking orbit velocity magnitude.
WE	6	R	Vector of target weights.
XMM	1	R	Mean motion of s/c in parking orbit.
PRO	1	R	Radial distance at injection.
PINC	1	R	Geocentric ecliptic inclination at injection
PTO	1	R	Time of injection

f) CØMMØN/TØP2/TOPSEP Parameters - Integer Variables

Name	Dimensions	Type	Definition
INACTV	20	I	Vector denoting which controls are active, on bounds, or within bound tolerance regions.
INSG	1	I	Flag set when S and G are input through namelist.
ITERAT	1	I	Iteration counter (in grid mode ITERAT indicates the index of the control being changed for a grid trajectory).
IWATE	1	I	Flag designating the desired control weighting scheme.
JMAX	1	I	Number of mission thrust phases.
JWATE	1	I	Flag designating target weighting.
KMAX	1	I	Number of thrust controls (THRUST (I,J)) chosen to be elements in U.
KØNVRJ	1	I	Convergence flag.
LØCCDC	1	I	Blank common location for storage of the inner products of the weighted sensitivity matrix columns.
LØCCM	1	I	Blank common location for storage of the magnitude of the weighted sensitivity column vectors.
LØCDU	1	I	Blank common location of the total control correction vector (not scaled by GAMA).
LØCDU1	1	I	Blank common location of the performance control correction vector (not scaled by GAMA).
LØCDU2	1	I	Blank common location of the constraint control correction vector (not scaled by GAMA).
LØCRFM	1	I	Blank common location of the s/c masses evaluated at event times for the reference and all trial trajectories in a single iteration.

Name	Dimensions	Type	Definition
LØCSDU	1	I	Blank common storage location for the original control correction vectors when a number of controls must be dropped during an iteration.
LØCSI*	1	I	Blank common location of the pseudo inverse of the weighted sensitivity matrix.
LØCSWG	1	I	Blank common storage location for the original weighted performance gradient when a number of controls must be dropped during an iteration.
LØCSWS	1	I	Blank common storage location for the original weighted sensitivity matrix when a number of controls must be dropped during an iteration.
LØCTS	1	I	Blank common location of event times for the reference and all trial trajectories in a single iteration.
LØCUL	1	I	Blank common location of minimum and maximum control bounds.
LOCWG*	1	I	Blank common location of the weighted performance gradient.
LØCWS*	1	I	Blank common location of the weighted sensitivity matrix.
LØCWU	1	I	Blank common location of the control weights.
LØCXR	1	I	Blank common location of the 6-component state vectors associated with the event times of the reference and all the trial trajectories of a single iteration.
MIN	1	I	Index on the scale factor in the GAMA vector which provides the best control correction.

*May be in compressed form if controls have been dropped during the iteration.

Name	Dimensions	Type	Definition
MPRINT	10	I	Flag designating TOPSEP print options.
NMAX	1	I	Maximum number of iterations.
NT	1	I	Number of targets.
NTNP	120	I	Vector of primary bodies associated with the event times of the reference and all trial trajectories in a single iteration.
NTPH	20	I	Vector of control phase numbers associated with the event times of the reference and all trial trajectories in a single iteration.
NTR	1	I	Trial trajectory counter (NTR=1 indicates the iteration reference trajectory).
NTYPE	1	I	Flag designating the type of control correction to be made during an iteration.
NU	1	I	Number of controls.
INJLØC	1	I	Index locating the selected injection controls in the U vector.

g) Common/TUG/Tug ΔV Parameters

Name	Dimensions	Type	Definition
AZMAX	1	R	Maximum launch azimuth constraint
AZMIN	1	R	Minimum launch azimuth constraint
RP1	1	R	Inner parking orbit radius
TGFUEL	1	R	Full capacity of tug stage
TUG	1	L	Flag controlling injection computations
TUGISP	1	R	Specific impulse of tug stage
TUGWT	1	R	Dry weight of tug stage

2.6.3 GØDSEP Labeled Commons

GØDSEP labeled commons were created following two specific guidelines as much as possible -- organization first by variable function, and second by variable type. Organization by function will hopefully simplify understanding of the program and minimize the number of common blocks required for any given subroutine. Organization by type is to facilitate conversion to machines which require double precision for many real variables, or which merely allocate different numbers of bytes of core for real, integer or logical variables.

Any variable for which further descriptions may be found under input description is denoted "(See Input)" and refers to Reference 1, Volume II (User's Manual) Section 2.3.

a) Common/DATAGI/Integer Variables Required Only for DATA Overlay

Name	Dimension	Type	Definition
CØNRD	1	L	Used for input only =F, set a priori control equal to a priori knowledge =T, assume a priori control is read in namelist $GØDSEP
IAUG	50	I	Parameter augmentation control (see Input)
IGFØRM	1	I	=0, input control uncertainties packed =1, input control uncertainties unpacked (see Input)
IPFØRM	1	I	=0, input knowledge uncertainties packed =1, input knowledge uncertainties unpacked (see Input)
MAXAUG	1	I	Maximum length allowed for augmented state vector (including S/C state) allowable maximum governed only by available core and dimensioned lengths of LIST (see Common/DIMENS/) and AUGLAB (see Common/LABEL/) arrays

Name	Dimension	Type	Definition
MAXDIM	5	I	Maximum allowable input dimensions on individual state vector partitions corresponding to actual dimensions of covariance partitions in subroutine NMLIST MAXDIM(1) = 6 (S/C state) MAXDIM(2) = 10 (solve-for parameters) MAXDIM(3) = 13 (dynamic consider) MAXDIM(4) = 15 (measurement consider) MAXDIM(5) = 10 (ignore)
XLAB	50	H	Parameter Hollerith labels corresponding to parameters as ordered for IAUG (see Input, IAUG)

b) Common/DATAGR/Real variables required only for DATA overlay

Name	Dimension	Type	Definition
DØPCNT	1	R	Average number of doppler (range-rate) measurements taken per day during tracking arcs (see Input)
SIGRS	1	R	Standard deviation in spin radius for equivalent station location errors (see Input)
SIGLØN	1	R	Standard deviations in longitude for equivalent station location errors (see Input)
SIGZ	1	R	Standard deviation in z-height for equivalent station location errors (see Input)
CØRLØN	1	R	Station-to-station longitude correlation for equivalent station location errors (see Input)

c) Common/DIMENS/Covariance dimensions and sub-block locators

Name	Dimension	Type	Definition
LIST	30	I	List of parameters included in augmented state vector in the order in which they appear in the covariance. LIST is used for locating elements of covariance and transition matrices where necessary. All parameters augmented are denoted by parameter number used for Input (see IAUG in Input). S/C state components - $x,y,z,\dot{x},\dot{y},\dot{z}$ - are denoted by -1,-2,-3,-4,-5,-6, respectively.
LISTDY	20	I	List of dynamic parameters included in transition matrices read from STM file. Parameter numbering and ordering conventions are the same as for LIST (above).

Name	Dimension	Type	Definition
LØCAUG	5x5	I	Array of locations of first word of covariance partitions within complete augmented covariance matrix. For example, since covariance blocks are ordered, S/C state, solve-for parameters, dynamic consider, measurement consider, ignore parameters, --LØCAUG(1,3) locates the first word of the sub-block of correlations between the S/C state and the dynamic consider parameters.
LØCBLK	5x5	I	Used for locating first word of covariance partitions when sub-blocks are stored separately but contiguously in core (for further explanation see AUGCNV Sec 3.3.1 and PPAK Sec 3.3.31)
LØCLAB	5	I	Locates within LIST and AUGLAB arrays the beginning of the parameter (LIST) or label (AUGLAB) lists for the five augmented state vector partitions (1) = 1 (2) = beginning of solve-for parameters (3) = beginning of dynamic consider parameters (4) = beginning of measurement consider parameters (5) = beginning of ignore parameters
NAUG	1	I	Dimension of augmented state vector
NAUGSQ	1	I	Total number of elements in augmented covariance matrix (=NAUG**2)
NBLK	1	I	Total number of elements required to store individual, packed covariance partitions (for further explanation, see AUGCNV, Sec 3.3.1, and PPAK, Sec 3.3.31)
NDIM	5	I	Dimensions of individual state vector partitions (1) = S/C state (2) = solve-for parameters (3) = dynamic consider parameters (4) = measurement consider parameters (5) = ignore parameters
NPHSTM	1	I	Number of dynamic parameters (including S/C state) used included in state transition matrices on STM file.

Name	Dimension	Type	Definition
d) Common/GUIDE/Guidance Related Variables Not Specifically Used for Scheduling or Propagation			
BURNP	4	R	Guidance interval parameters (1) - vehicle mass at guidance start (2) - thrust acceleration magnitude at guidance start (3) - vehicle mass at guidance end (4) - thrust acceleration magnitude at guidance end
CØNWT	5	R	Control weighting factors, following correspondences assumed (1) - acceleration magnitude (2) - cone angle (3) - clock angle (4) - cutoff time (5) - startup time
DELAY	1	R	Guidance delay time for current maneuver
S	6x5	R	Guidance sensitivity matrix of S/C state at cutoff time with respect to controls
SMAT	15	R	Sensitivity matrix of target parameters w.r.t. control parameters
TARWT	3	R	Target parameter weights
TBURN	1	R	Length of burn interval for current guidance maneuver
TGSTØP	1	R	Stop time for integrator if either guidance or prediction requires integration of transition matrices to some time past TFINAL. For both guidance and prediction TDUR (Common/TIME/) is defined according to the maximum of TGSTØP and TFINAL
TØFF	1	R	Cutoff time for current guidance maneuver
TØN	1	R	Execution time for current guidance maneuver
UMAX	5	R	Maximum (1σ) control corrections allowed
VARDV	4	R	Array of variances of delta-V execution error parameters (1) - magnitude proportionality ($100\%^2$) (2) - magnitude resolution (km^2/s^2) (3) - in-ecliptic pointing (rad^2) (4) - out-of-ecliptic pointing (rad^2)
VARMAT	18	R	Variation matrix, sensitivity of target conditions with respect to S/C state at cutoff time
IPØL	1	I	Guidance policy flag for current guidance event (see IGPØL, Input)
IREAD	1	I	Read policy for namelist $GEVENT for current guidance event (see IGREAD, Input)
NCØN	1	I	Number of controls to be used for low thrust guidance

e) Common/KEPCØN/Transformations Required When Ephemeris Body State is in Keplerian Elements

Name	Dimension	Type	Definition
DXDKAF	36	R	Transformation from Keplerian to cartesian elements for ephemeris body evaluated at cutoff time of guidance event
DXDKBR	36	R	Transformation from Keplerian to cartesian elements for ephemeris body evaluated at guidance maneuver execution time
DXDKST	36	R	Transformation from Keplerian to cartesian elements for ephemeris body evaluated at time TSTM, the current trajectory time as defined by the STM file
LISTPH	6	I	List of ephemeris parameter numbers for whichever set (Keplerian or cartesian) is augmented to the S/C state

f) Common/LABEL/Labeling Arrays

Name	Dimension	Type	Definition
AUGLAB	30	H	Array of parameter labels, AUGLAB(I) contains a six-character Hollerith label which corresponds to the parameter number in LIST(I) (see LIST, Common/DIMENS/)
EVLAB	2x5	H	Array of event labels (1,1),(2,1) - propagation (1,2),(2,2) - eigenvector (1,3),(2,3) - thrust (1,4),(2,4) - guidance (1,5),(2,5) - prediction
JØBLAB	10	H	Run identifying label input through namelist $GØDSEP and printed at the top of the first page of each measurement and event print
MESLAB	2x10	H	Array of measurement labels used for printing in MEASPR (see MEASPR, sec. 3.3.22 for further details)
PGLAB	5x5	H	Array of labels for control covariance subblocks, used primarily for punching. Upper triangle elements are identical to those names used for control uncertainty input (CXSG,CXUG etc). Lower triangle blocks correspond to transposes of upper triangle blocks -- their labels are so denoted by an added dollar sign (CXSG$,CXUG$, etc).

Name	Dimension	Type	Definition
PLAB	5x5	H	Array of labels for knowledge covariance sub-blocks. Upper triangle elements are identical to those names used for knowledge uncertainty input (CXS, CXU, etc). Lower triangle blocks correspond to transposes of upper triangle blocks -- their labels are so denoted by an added dollar sign (XS$, CXUS, etc).
VECLAB	2x5	H	Array of word labels for augmented state vector partitions (1,1),(2,1) - state (1,2),(2,2) - solve-for (1,3),(2,3) - dynamic (1,4),(2,4) - measurement (1,5),(2,5) - ignore

g) Common/LØCATE/Parameters Used To Locate Matrices In Blank Common

P	1	I	Location of current knowledge covariance in blank common
PG	1	I	Location of current control covariance in blank common, if guidance events are included
PWLS	1	I	Location of weighted least squares reference covariance in blank common if using sequential weighted least squares OD algorithm
PHI	1	I	Location of complete augmented transition matrix in blank common if not using covariance integration option
PTEMP	1	I	Location in blank common of temporary working area the size of the augmented covariance (and therefore transition matrix, also) By convention the output of CØVP is always located by PTEMP
PLØCAL	1	I	Location in blank common of local working storage area the size of the augmented co-variance matrix. This area is intended to be used locally within a subroutine and not to be saved for use in another subroutine.

Name	Dimension	Type	Definition
H	1	I	Location of observation matrix in blank common
GAIN	1	I	Location of gain matrix in blank common
PG1	1	I	Locations of four augmented covariance size blocks in blank common used for guidance computations
PG2	1	I	

h) Common/LØGIC/Logical Variables

CHEKPR	10	L	Array of flags controlling checkout print options (see Input)
DYNØIS	1	L	Flag controlling computation of effective process noise =.TRUE., compute effective process noise =.FALSE., do not compute effective process noise
GAINCR	1	L	Flag controlling creation of GAIN file (TAPE 4) =.TRUE., create GAIN file =.FALSE., do not create GAIN file
GENCØV	1	L	Flag indicating if current run is generalized covariance run =.TRUE., generalized covariance run =.FALSE., not generalized covariance run
MESH	1	L	Flag indicating if scheduled trajectory time can be meshed with some time print on the STM file within specified forward and backward tolerances (TØLFØR,TØLBAK,common/PRØPR/) =.TRUE., meshing successful =.FALSE., meshing not successful
PDØT	1	L	Flag controlling covariance propagation =.TRUE., propagate by integration of covariance variational equations =.FALSE., propagate by state transition matrices

Name	Dimension	Type	Definition
PRINT	1	L	Flag controlling measurement print =.TRUE., causes full print before and after current measurement =.FALSE., suppresses measurement print except for that on SUMMARY file if summary print requested (see SUMARY, common/LØGIC/)
PRNCØV	5	L	Array of flags controlling print options on covariance sub-blocks (see Input)
PRNSTM	5	L	Array of flags controlling print options on transition matrix partitions (see Input)
PRØPG	1	L	Flag controlling propagation of control covariance =.TRUE., propagate control simultaneously with knowledge covariance =.FALSE., do not propagate control covariance
PUNCHE	5	L	Array of flags controlling punching of complete augmented state uncertainties for different event types (see Input).
SCHFTL	1	L	Flag controlling termination or continuation of run after mesh failure on STM file if MESH = .TRUE., SCHFTL has no effect. if MESH = .FALSE., then SCHFTL = .TRUE., will terminate error analysis processing, while SCHFTL = .FALSE., will result in diagnostic print and the currently scheduled measurement or event will not be processed
SUMARY	1	L	Flag controlling SUMMARY file print =.TRUE., prints summary information for all measurements on SUMMARY file (TAPE 8) =.FALSE., no summary print
VRNIER	1	L	Flag indicating if current guidance event is a vernier (=.TRUE.) or a primary (=.FALSE.)

---

i) Common/MEASI/Measurement Related Integer Variables

Name	Dimension	Type	Definition
IAUGPH	1	I	Parameter number of first ephemeris element as used for input (See IAUG,Input).

Name	Dimension	Type	Definition
IAUGST	1	I	Parameter number for first station location parameter
IBAZEL	1	I	Parameter number for first azimuth-elevation angle bias parameter
IBDIAM	1	I	Parameter number for apparent planet diameter measurement bias
IBRAD	1	I	Parameter number for right ascension/declination measurement biases
IBSTAR	1	I	Parameter number of first star-planet angle measurement bias
IB2WAY	1	I	Parameter number of first 2-way DSN measurement bias term
IB3WAY	1	I	Parameter number of first 3-way DSN measurement bias term
IDATYP	1	I	Leading digit of decoded measurement type =1, ground-based range-rate =2, ground-base range =3, azimuth-elevation angles =4, on-board optics - star-planet angle =5, on-board optics - apparent planet diameter
IDMAX	1	I	Maximum number allowed to be assigned to a dynamic parameter. All parameter numbers less than or equal to IDMAX are assumed to correspond to dynamic parameters. Those greater than IDMAX are assumed to be measurement parameters.
IEPHEM	1	I	If any ephemeris elements included in augmented state vector, denotes form =0, time-evolving cartesian =1, stationary cartesian =2, stationary Keplerian
IGAIN	1	I	Flag indicating gain computation algorithm to be used (see Input)
ISTA1 ISTA2 ISTA3	1 1 1	I I I	Parameters used in decoding measurement codes. For further explanation see ØBSERV, sec. 3.3.26.
MAXSTA	1	I	Maximum number of stations for which station location errors and range and range-rate biases can be augmented to the state (maximum number accommodated by IAUG array). See ØBSERV, sec. 3.3.26 for further explanation.

Name	Dimension	Type	Definition
NEPHEL	1	I	Number of ephemeris elements augmented to state for current error analysis run
NR	1	I	Dimension of observation vector for measurement currently being processed
NSØLVE	1	I	Total number of variables and parameters being estimated by OD algorithm (number of S/C state variables plus number of solve for parameters)
NST	1	I	Total number of ground stations defined in STALØC array for possible use in ground-based observations (maximum 9). For further explanation see NST and STALØC, in Input.

j) Common/MEASR/Measurement Related Real Variables

Name	Dimension	Type	Definition
AZMUTH	1	R	Azimuth angle in degrees from station ISTA1 (ØBSERV, sec 3.3.26) computed only for azimuth-elevation angle measurements
AZMTH2	1	R	Azimuth angle in degrees from station ISTA2 (ØBSERV, sec 3.3.26) computed only for azimuth-elevation angle measurements and if ISTA2 > 0.
BDYDEC	1	R	Declination angle of the target body (in degrees) as seen from the designated observation
BDYRTA	1	R	Right ascension angle of the target body (in degrees) as seen from the designated observatory
ELEV	1	R	Elevation angle in degrees from station ISTA1 (ØBSERV, sec 3.3.26) computed for all ground-based measurements
ELEV2	1	R	Elevation angle in degrees from station ISTA2 (ØBSERV, sec 3.3.26) computed for all ground-based measurements when ISTA2 > 0
R	16	R	Dual purpose measurement noise matrix. Before the knowledge covariance is updated at a measurement, R is the covariance of the measurement white noise. After the knowledge covariance is updated, R is the measurement residual matrix. For further explanation see Vol. I, Analytical Manual, sec 6.4.
RANGE	1	R	Range in km from station ISTA1 (ØBSERV, sec 3.3.26) computed for all ground-based measurements

Name	Dimension	Type	Definition
RANGE2	1	R	Range in km from station ISTA2 (ØBSERV, sec 3.3.26) computed for all ground-based measurements if ISTA2 > 0
RRATE	1	R	Range-rate in km/s from station ISTA1 (ØBSERV, sec 3.3.26) computed for doppler (range-rate) measurements only
RRATE2	1	R	Range-rate in km/s from station ISTA2 (ØBSERV, sec 3.3.26) computed for doppler (range-rate) measurements only, and only if ISTA2 > 0
SCDEC	1	R	S/C geocentric equatorial declination in degrees, computed for all ground-based measurements
SCGLØN	1	R	S/C geocentric equatorial longitude in degrees, computed for all ground-based measurements
STALØC	3x9	R	Array of station locations in cylindrical equatorial coordinates STALØC (1,I) = spin radius (km) STALØC (2,I) = longitude (degrees externally, radians internally) STALØC (3,I) = height (km) (See Input)
STARDC	3x9	R	Array of ecliptic star direction cosines (or, equivalently, unit vectors in star directions) See Input
STPANG	3	R	Array of star-planet angle measurements in degrees, computed only for star-planet angle measurements. (1)-angle between planet/target body and star ISTA1 (ØBSERV, sec 3.3.26) (2),(3) - same as (1) above only for stars ISTA2 and ISTA3 respectively
VARMES	10	R	Array of measurement white noise variance. Default values and input are by standard deviations in array SIGMES (see Input) internal values require units conversion as well as squaring. (1), 2-way doppler (km^2/s^2) (2), 2-way range (km^2) (3), 3-way equivalent frequency drift (km^2/s^2) (4), 3-way range (km^2) (5), azimuth angle (rad^2)

Name	Dimension	Type	Definition
			(6), elevation angle (rad^2) (7), on-board optics-star-planet angle (rad^2) (8), on-board optics-apparent planet diameter (rad^2) (9), on-board optics-center finding uncertainty, in conjunction with star-planet angle (rad^2) (10), not used

k) Common/PRØPI/Propagation Related Integer Variables

Name	Dimension	Type	Definition
IPRØP	1	I	Flag controlling print options with propagation event =0, no print =1, print standard deviations and correlation coefficients for S/C state vector only =2, full eigenvector print
LAFTER	1	I	not used
LBURN	1	I	not used
LDELAY	1	I	not used

l) Common/PRØPR/Propagation Related real Variables

Name	Dimension	Type	Definition
EPTAU	3x2	R	Array of correlation times for thruster process noise terms; EPTAU(I,J) represents correlation time for process whose variance is EPVAR(I,J) (See Below)
EPVAR	3x2	R	Array of variances for thruster noise processes. All elements are used for covariance integration, while only elements EPVAR(I,1) are used in the effective process noise model. Primary processes (1,1), magnitude variance (2,1), cone angle pointing variance (3,1), clock angle pointing variance Secondary processes (1,2), magnitude variance (2,2), cone angle pointing variance (3,2), clock angle pointing variance
GMASS	1	R	not used

Name	Dimension	Type	Definition
GTBURN	3x3	R	GT matrix (See DYNØ, Section 3.3.10) evaluated at the beginning of a guidance burn interval.
GTDLAY	3x3	R	GT matrix (See DYNØ, Section 3.3.10) evaluated at cutoff time of guidance interval.
GTSAVE	3x3	R	GT matrix (See DYNØ, Section 3.3.10) saved at beginning of each propagation interval during normal knowledge propagation.
Q	6x6	R	Effective process noise matrix computed in DYNØ (Section 3.3.10).
SAVACC	3	R	Thrust acceleration magnitude for bias, and first and second noise processes.
TG	1	R	Input epoch for control uncertainties if different from epoch for knowledge uncertainties.
TØLBAK	1	R	Backward tolerance on reading transition matrices from STM file.
TØLFØR	1	R	Forward tolerance on reading transition matrices from STM file.
XG	6	R	not used.

m) Common/SCHEDI/Scheduling Related Integer Variables

Name	Dimension	Type	Definition
IGPØL	20	I	Array of guidance policy control flags =0, no maneuver, print control uncertainties =1, target to cartesian state, XYZ, at time specified by TIMFTA =2, two variable B-plane targeting (B·T, B·R) =3, three variable B-plane targeting (B·T, B·R, T_{SOI}) =4, closest approach targeting (radius of closest approach, inclination, time of closest approach). =5, XYZ targeting, variable time of arrival.
IGREAD	20	I	Array of guidance event read control flags. (See Input)
ITPØL	20	I	Not used.

Name	Dimension	Type	Definition
MCØDE	50	I	Array of measurement (and propagation event) codes used in scheduling (See SCHED, Section 3.3.36).
MCØUNT	1	I	Measurement counter, total cumulative number of measurements processed.
MESEVN	1	I	Current measurement or event code.
MNEXT	1	I	Code for measurement (or propagation event) to be scheduled after the current event.
MPCNTR	11	I	Array of counters for classes of data types used for measurement print control (See Input).
MPFREQ	11	I	Array of print frequencies for measurement print control (See Input).
NCNTE	1	I	Counter indicating number of current (or most recently executed) eigenvector event.
NCNTG	1	I	Counter indicating number of current (or most recently executed) guidance event.
NCNTP	1	I	Counter indicating number of current (or most recently executed) prediction event.
NCNTT	1	I	Counter indicating number of current (or most recently executed) thrust event.
NEIGEN	1	I	Total number of eigenvector events to be processed.
NGUID	1	I	Total number of guidance events to be processed.
NPRED	1	I	Total number of prediction events to be processed.
NSCHED	1	I	For input, number of scheduling cards to be read. During execution, number of elements of SCHEDM (common/SCHEDR/) to be tested for scheduling.
NTHRST	1	I	Total number of thrust events to be processed.

Name	Dimension	Type	Definition
n) Common/SCHEDR/Scheduling Related real Variables			
DELTIM	1	R	Propagation interval length, time between previously and currently scheduled event. DELTIM computed between STM file time when reading STM file, and between actual scheduled times for PDØT and STM file generation.
SCHEDM	3x50	R	Array of measurement schedule times. SCHEDM(1,I) = Next time to be scheduled for measurement type MCØDE(I) SCHEDM(2,I) = Stop time for MCØDE(I) SCHEDM(3,I) = Time increment for scheduling MCØDE(I).
TCURR	1	R	Current trajectory time.
TCUTØF	20	R	Array of guidance event cutoff times.
TDELAY	20	R	Array of guidance event delay times.
TEIGEN	20	R	Array of eigenvector event times.
TFINAL	1	R	Final trajectory time for current run.
TGUID	20	R	Array of guidance event times.
TIMFTA	1	R	Target condition evaluation time for fixed time of arrival targeting.
TMNEXT	1	R	Time of next measurement (or propagation event) to be scheduled (See SCHED, Section 3.3.36).
TPAST	1	R	Time of most recently scheduled measurement or event. Set to previous scheduled time when generating STM file or executing PDØT. Set to previous STM file time when reading from STM file.
TPRED	10	R	Array of prediction event times.
TPRED2	10	R	Array of times predicted to for prediction events.

Name	Dimension	Type	Definition
TSTM	1	R	Current time from STM file when reading STM file.
TTHRST	20	R	Array of thrust event times.

2.6.4 SIMSEP Common Blocks

The SIMSEP overlay of MAPSEP has seven common blocks: DYNØS, ISIM1, ISIM2, SIM1, SIM2, SIMLAB and STØREC. DYNØS contains the random number seed and thrust noise terms; it is essential to all SIMSEP routines that call the random number generator, RNUM. SIM1 and ISIM1 are common blocks containing information essential to the operation of SIMSEP and execution of the Monte Carlo loop. SIM1 contains real data and ISIM1, integer data. SIM2 and ISIM2 have a correspondence similar to SIM1 and ISIM1 and contain accumulated statistical data. SIMLAB contains Hollerith labels used throughout the program. Finally, STØREC is a storage common block with three sets of data, each pertaining to the actual, estimated, and reference world integrating conditions.

a) Common/DYNØS/Process Noise Variables

Name	Dimension	Type	Definition
IRAN	1	I	Random number seed.
TVERR	6x3	R	Time varying thrust errors.

b) Common/ISIM1/SIMSEP Integer Variables

Name	Dimension	Type	Definition
IEPH	5	I	Ephemeris planet code.
IGL	5	I	Guidance Flag.
INREF	1	I	State vector read-in flag.
IØUT	1	I	Printout frequency flag.
IPUNCH	1	I	Punch output flag.
ITMX	5	I	Maximum number of iterations allowed in non-linear guidance.

Name	Dimension	Type	Definition
JMAX	1	I	Number of the last active thrust control phase.
JMIN	1	I	Number of the first active thrust control phase.
KDIM	5	I	Dimension of the augmented knowledge covariance.
KTERR	5	I	Option flag for calculating target errors after a guidance correction.
LSTAR	6x5	I	List of target variable codes.
MTPH	5	I	Thrust phase number at a guidance event.
NCYCLE	1	I	Number of Monte Carlo cycles.
NEP2	2	I	Active ephemeris planet code.
NGUID	1	I	Number of the guidance event
NTAR	5	I	Number of target variables.
NTC	5	I	Number of control variables.
NTPL	5	I	Number of the target planet.

c) Common/ISIM2/Monte Carlo Integer Variables

Name	Dimension	Type	Definition
KATH	1	I	Dimension of the ATHCØV covariance matrix (see Common SIM2).
MC	1	I	Number of Monte Carlo cycles executed previously.
NSAMP	5	I	Number of Monte Carlo cycles executed previously for a given guidance event.

d) Common/SIMLAB/SIMSEP Labels

Name	Dimension	Type	Definition
LABCON	12x5	I	Stores Hollerith data pertaining to control variables.
LABTAR	12x5	I	Store Hollerith data pertaining to target variables.
NAMEX	12	I	Store Hollerith state vector labels.

e) Common/SIMI/Trajectory Simulation Real Variabl

Name	Dimension	Type	Definition
AØK	1	R	Backup convergence tolerance for the weak convergence test.
CØNWT	6x5	R	Control weights.
CPMAX	1	R	Computer processing time limit.
DVMDØT	1	R	Mass flow rate for chemical propulsion system.
DVMXN	1	R	Maximum delta-velocity magnitude step.
EPMERR	6x7x2	R	Ephemeris error covariances (in eigenvector/ eigenvalue format) for the ephemeris bodies.
EXVERR	4	R	Midcourse velocity correction execution errors.
GMERR	3	R	Gravitational constants errors.
MEND	1	R	S/C reference mass at TEND.
PG	6x7	R	Spacecraft control error matrix (eigenvector/ eigenvalue format).
RMGE	5	R	S/C reference mass at a guidance event.
RMTAR	5	R	S/C reference mass at a target point.
RXGE	6x5	R	Reference state vector at a guidance event.
RXTAR	6x5	R	Reference state vector at a target point.
SCERR	10	R	Spacecraft errors.
SMAT	36x5	R	Sensitivity or guidance matrix.
SPFIMP	1	R	Specific impulse for chemical propulsion system.
TCERR	6x20	R	Thrust bias errors.
TEPH	2	R	Epoch of evaluation of the ephemeris errors.
TGE	5	R	Guidance event epoch
TØL	5	R	Target condition tolerances.

Name	Dimension	Type	Definition
TTAR	5	R	Target epoch.
UNTAR	6x5	R	Conversion factor for converting target variables from internal to external printout units.
XEND	6	R	Reference state vector at TEND.
XEPH	6x2	R	Ephemeris planet state vector.
XTARG	6x5	R	Reference trajectory target variables at TTAR.

f) Common/SIM2/Monte Carlo Real Variables

Name	Dimension	Type	Definition
ADVT	2	R	Total delta-velocity magnitude statistics.
AMASS	2	R	Accumulated final spacecraft mass statistics.
ATHCØV	420	R	Accumulated total thrust control statistics.
CNCØV	42x5	R	Accumulated active thrust control error statistics.
DVCØV	3x4x5	R	Accumulated delta-velocity vector error matrix.
DVMAGS	2x5	R	Accumulated delta-velocity magnitude statistics.
ENDCØV	6x7	R	Spacecraft control error covariance at the final trajectory time TEND.
GCCØV	6x7x6	R	Accumulated spacecraft control error statistics evaluated at guidance events.
GMCØV	2x5	R	Accumulated mass error statistics evaluated at guidance events.
TCCØV	6x7x5	R	Accumulated spacecraft control error statistics evaluated at the target points.
TERCØV	42x5	R	Accumulated target error statistics.
TMCØV	2x5	R	Accumulated mass error statistics evaluated at target points.

g) Common/STØREC/Stored Variables

Name	Dimension	Type	Definition
SCRA1	1	R	Stored radiation pressure coefficient.
SECC1	4x10	R	Stored planetary eccentricities.
SEMN1	15	R	Stored lunar orbital elements.
SEXV1	1	R	Stored exhaust velocity.
SINC1	4x10	R	Stored planetary inclinations.
SMEAN1	4x10	R	Stored planetary mean anomalies.
SNTPH1	1	R	Stored thrust phase number.
SØMEG1	4x10	R	Stored planetary nodes.
SØMGT1	4x10	R	Stored planetary longitude of APSES.
SPM1	11	R	Stored planetary masses.
SPO1	1	R	Stored electric power constant.
SSAX1	2x10	R	Stored planetary semi-major axes.
SSCM1	1	R	Stored S/C mass.
SSM1	1	R	Stored solar mass.
STEFF1	1	R	Stored thruster efficiency.
STHRT1	6x20	R	Stored thrust control profile.
SXEPH1	6x2	R	Stored ephemeris body cartesian states.

Note that there are, in fact, three sets of data in STØREC corresponding to post-scripts, 1, 2, and 3. For example, SCRA1 contains the radiation pressure coefficient used while integrating an actual trajectory. SCRA2 also contains a radiation pressure coefficient but is used while integrating an estimated trajectory. Likewise, SCRA3 and all post-script-3 constants are used for generating the reference trajectory.

3.1 Subroutine: MAPSEP

Purpose: MAPSEP is the executive routine that selects the mode of operation (primary overlay): TØPSEP, GØDSEP, SIMSEP, or REFSEP. In addition, MAPSEP calls a fifth primary, overlay DATAM, to initialize many trajectory parameters, and to print the initial trajectory information.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
MØDE	I	C	Flag determines the program's operational mode. = ± 1, Targeting and Optimization (TØPSEP). = ± 2, Error analysis (GØDSEP). = ± 3, Simulation (SIMSEP). = ± 4, Reference trajectory propagation (REFSEP). Positive values will cause recycling back to the MAPSEP main, while negative numbers will cause recycling back to the mode main.
ICYCLE	O	C	Flag used for writing the mode's namelist onto disc when recycling back to the mode's main. = 0, Do not store the namelist variables on disc. = 1, Store the namelist variables on disc.
INIT	O	C	Flag used to read namelist $TRAJ from disc during recycling.

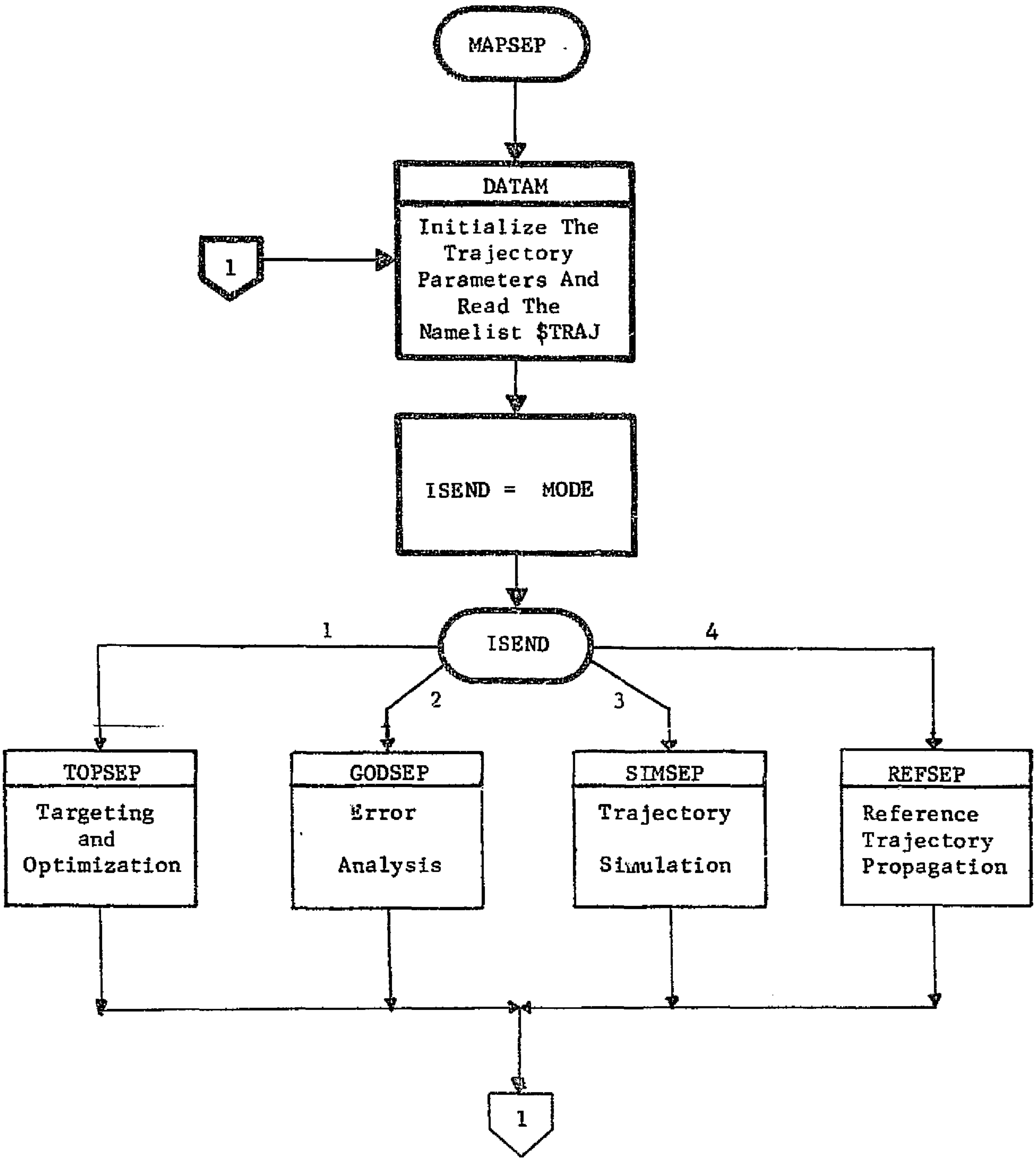
Local Variables:

Variable	Definition
ISEND	Index used to select the program's mode of operation. ISEND is the absolute value of MØDE.

Subroutines Called: DATAM, TØPSEP, GØDSEP, SIMSEP, REFSEP

Common Blocks: (BLANK), CØNST, CYCLE, EDIT, EPHEM, TIME, TRAJ1, TRAJ2, TRKDAT, WØRK

Logic Flow:



3.1.1 Subroutine: BLKDAT

Purpose: To initialize default values of program constants.

Method: DATA statements.

Remarks: The following four pages contain a listing of BLKDAT with respect to the default constants in MAPSEP. The variables are defined in appropriate common blocks (Section 2.6).

Common CØNST: AU, PI, RAD, TM, FØP, BIG, SMALL

Common EPHEM: DJ1900, SUN, PLANET, SMASS, PMASS, CSAX, CECC, CINC, CØMEG, CØMEGT, CMEAN, EMN, SPHERE, SRADIS, PRADIS

Common TRAJ1: UP, VP

```
      DATA AU/1.49597893E8/
      DATA BODY/12*6H      /
      DATA DJ1900/2415020.0/
      DATA PI,RAD/3.14159265358979323846,57.29577951308232/
      DATA TM/86400.0/
      DATA SMALL,BIG,FOP,FOV/1.E-20,1.E20,1.E-15,1.E-25/
      DATA SUN,PLANET/6HSUN   ,6HMERCRY,6HVENUS ,6HEARTH ,6HMARS  ,
     $ 6HJUPITR,6HSATURN,6HURANUS,6HNEPTNE,6HPLUTO ,6HENCKE ,6HMOON  /
      DATA UP(1,1),UP(2,1),UP(3,1),VP(1,1),VP(2,1),VP(3,1)/6*0.0/




      DATA SMASS,PMASS/
     S              1.32712499E11          ,
     M              2.2181597693464 72E+04,
     V              3.2486010300546 70E+05,
     E              4.0350397886774 69E+05,
     A              4.2828443863555 56E+04,
     J              1.2670771883808 76E+08,
     S              3.7926525773203 8E+07,
     U              5.7877234627125 86E+06,
     N              6.8905762720664 44E+06,
     P              7.3240893487858 59E+04,
     X              1.0                    ,
     M              4.8983099709676462E3
     $/

          SEMIMAJOR AXIS OF ORBIT (KM)

      DATA CSAX/
     M              5.790913494324970E+07,     0.                    ,
     V              1.082088833003187E+08,     0.                    ,
     E              1.495979274075146E+08,     0.                    ,
     A              2.279410379820059E+08,     0.                    ,
     J              7.783283664940758E+08,     0.                    ,
     S              1.426990814457794E+09,     0.                    ,
     U              2.869628820533920E+09,    -8.528276684144000E+04,
     N              4.496471798589325E+09,     1.810382837802377E+05,
     P              5.890213765146733E+09,     0.                    ,
     X              0.                   ,     0.
     $/

          ECCENTRICITY OF PLANET ORBIT

      DATA (CECC(I),I=1,20)/
     M              2.056142100000000E-01,     2.046000000000000E-05,
     M             -3.000000000000000E-08,     0.                    ,
     V              6.820690000000000E-03,    -4.774000000000000E-05,
     V              9.100000000000000E-08,     0.                    ,
     E              1.675104000000000E-02,    -4.180000000000000E-05,
     E             -1.260000000000000E-07,     0.                    ,
     A              9.331290000000000E-02,     9.206400000000000E-05,
     A             -7.700000000000000E-08,     0.                    ,
     J              4.833760000000000E-02,     1.630200000000000E-04,
     J              0.                   ,     0.
```

```
$/
 DATA (CECC(I),I=21,40)/
S               5.5890000000000000E-02,      -3.4705000000000000E-04,
S               0.                    ,       0.                    ,
U               4.7046300000000000E-02,       2.7204000000000000E-04,
U               0.                    ,       0.                    ,
N               8.5284900000000000E-03,       7.7010000000000000E-05,
N               0.                    ,       0.                    ,
P               2.4880330536269240E-01,       0.                    ,
P               0.                    ,       0.                    ,
X               0.                    ,       0.                    ,
X               0.                    ,       0.
$/

     INCLINATION OF PLANET ORBIT

 DATA (CINC(I),I=1,20)/
M               1.2222332281833380E-01,       3.2477668497527890E-05,
M              -3.1997702953229400E-07,       0.                    ,
V               5.9230026790728640E-02,       1.7555103392976300E-05,
V              -1.6968478838833780E-08,       0.                    ,
E               0.                    ,       0.                    ,
E               0.                    ,       0.                    ,
A               3.2294408926068390E-02,      -1.1780972450961800E-05,
A               2.2010541122373030E-07,       0.                    ,
J               2.2841026958113520E-02,      -9.6962736221907140E-05,
J               0.                    ,       0.
$/
 DATA (CINC(I),I=21,40)/
S               4.3503786047002000E-02,      -7.7570188977525800E-05,
S               0.                    ,       0.                    ,
U               1.3486546981105070E-02,       9.6962736221907250E-06,
U               0.                    ,       0.                    ,
N               3.1053770716109040E-02,      -1.5998851476614700E-04,
N               0.                    ,       0.                    ,
P               2.9967069708596940E-01,       0.                    ,
P               0.                    ,       0.                    ,
X               0.                    ,       0.                    ,
X               0.                    ,       0.
$/

     LONGITUDE OF ASCENDING NODE OF PLANET ORBIT

 DATA (COMEG(I),I=1,20)/
M               8.2285195951788380E-01,       2.0685787738741190E-02,
M               3.0349336437457010E-06,       0.                    ,
V               1.3226043500275470E+00,       1.5705345274070970E-02,
V               7.1558499331767710E-06,       0.                    ,
E               0.                    ,       0.                    ,
E               0.                    ,       0.                    ,
A               8.5148403741548150E-01,       1.3456343088772030E-02,
A              -2.4240684055476850E-08,      -9.3084226773030820E-08,
J               1.7355180775297110E+00,       1.7644793923981550E-02,
J               0.                    ,       0.
```

```
$/
 DATA (COMEG(I),I=21,40)/
S                    1.968444580475854E+00,          1.523977869610149E-02,
S                    0.                   ,          0.                   ,
U                    1.282640770442747E+00,          8.912087492996046E-03,
U                    0.                   ,          0.                   ,
N                    2.280773383300414E+00,          1.923032858668217E-02,
N                    0.                   ,          0.                   ,
P                    1.914337550102258E+00,          0.                   ,
P                    0.                   ,          0.                   ,
X                    0.                   ,          0.                   ,
X                    0.                   ,          0.
$/
     LONGITUDE OF PERIGEE OF PLANET ORBIT

 DATA (COMEGT(I),I=1,20)/
M                    1.324699617794565E+00,          2.714840258929940E-02,
M                    5.143873155572180E-06,          0.                   ,
V                    2.271787456683804E+00,          2.457486612586557E-02,
V                   -1.704120089100021E-05,          0.                   ,
E                    1.766630813274085E+00,          3.000526416797356E-02,
E                    7.902463002085463E-06,          5.817764173314452E-08,
A                    5.833208058570250E+00,          3.212729365018996E-02,
A                    2.266503959167080E-06,         -2.084698828771005E-08,
J                    2.218562188703190E-01,          2.812302353243390E-02,
J                    0.                   ,          0.
$/
 DATA (COMEGT(I),I=21,40)/
S                    1.589799665616077E+00,          3.419861162136240E-02,
S                    0.                   ,          0.                   ,
U                    2.950242608382752E+00,          2.834608630711233E-02,
U                    0.                   ,          0.                   ,
N                    7.635293817954256E-01,          1.532704515870120E-02,
N                    0.                   ,          0.                   ,
P                    3.909919302791948E+00,          0.                   ,
P                    0.                   ,          0.                   ,
X                    0.                   ,          0.                   ,
X                    0.                   ,          0.
$/
     MEAN ANOMALY OF PLANET ORBIT

 DATA (CMEAN(I),I=1,20)/
M                    1.785111955351731E+00,          7.142471000792648E+02,
M                    8.726646259971626E-09,          0.                   ,
V                    3.710626171888563E+00,          2.796244623278380E+02,
V                    1.682497398922535E-06,          0.                   ,
E                    6.256583784118674E+00,          1.720196976768520E+02,
E                   -1.954768762233648E-07,         -1.221730476396035E-09,
A                    5.576840523254305E+00,          9.145887725994726E+01,
A                    2.365444735227922E-07,          4.363323129985823E-10,
J                    3.930858175721440E+00,          1.450191927757481E+01,
J                    0.                   ,          0.
```

```
$/
 DATA (CMEAN(I),I=21,40)/
S                 3.0620404062515 32E+00,        5.837120989790549E+00,
S                 0.                    ,        0.                   ,
U                 1.297162152226178E+00,         2.046547919058511E+00,
U                 0.                    ,        0.                   ,
N                 7.204851506367511E-01,         1.046371040833037E+00,
N                 0.                    ,        0.                   ,
P                 3.993890006707340E+00,         6.962635708298997E-01,
P                 0.                    ,        0.                   ,
X                 0.                    ,        0.                   ,
X                 0.                    ,        0.
$/
 DATA EMN/
O                 4.523601515           ,       -0.000192422           ,
O                 0.000036267           ,        0.000000034           ,
W                 5.63515154            ,        0.001944367           ,
W                -0.000180205           ,       -0.000000209           ,
L                 4.719965573           ,        0.229971481           ,
L                -0.000019774           ,        0.000000033           ,
I                 0.089804108           ,
E                 0.054900489           ,
A                 3.843984402E5
$/

 DATA SPHERE/
M                   3.189878022841E+05 ,
V                   1.458336566233E+06 ,
E                   2.167226195872E+06 ,
A                   1.563493250956E+06 ,
J                   7.664078431145E+07 ,
S                   9.394359395396E+07 ,
U                   1.010084500916E+08 ,
N                   1.677463630809E+08 ,
P                   4.831459314755E+07 ,
X                   0.0                ,
M                   8.555281139223E-06
$/
  DATA SRADIS,PRADIS/
S                   6.95992E+05         ,
M                   2.43500E+03         ,
V                   6.05000E+03         ,
E                   6.37816E+03         ,
A                   3.39340E+03         ,
J                   7.13720E+04         ,
S                   6.04010E+04         ,
U                   2.36500E+04         ,
N                   2.50020E+04         ,
P                   7.01600E+03         ,
X                   0.0                 ,
M                   1.73809E+03
$/
  RETURN
  END
```

3.1.2 Subroutine: DATAM

Purpose: To read input data and initialize trajectory and spacecraft parameters for all MAPSEP modes.

Method: After DATAM executes the default value initialization, the namelist $TRAJ is read. The dimensions and definitions for variables contained in this namelist are discussed in detail in Section 2.1 of the User's Manual. The input data are processed and stored in labeled common for subsequent use in any of the three possible modes. User options specified by input determine the degree of data preparation and the logic operations within the main cycle of the program.

Remarks: Some variables appearing in DATAM are initialized from the namelist with units specified in the User's Manual. Before these variables are stored in common, they are converted, if necessary, to internal units which are: kg, kw, km, sec, km/sec, and radians

Input/Output:

Variable	Input/ Output	Namelist/ Common	Definition
ACC (STEP)	I	N/C	Scaling factor of the integration step size.
BIG	O	C	Large constant, $1*10^{20}$.
BØDY	O	C	Hollerith names of bodies considered in integration.

Variable	Input/ Output	Namelist/ Common	Definition
BØDYIN	I	N	Input ephemeris data for body not included block data.
CECC	I/O	C	Array of orbital eccentricities and rates.
CINC	I/O	C	Array of orbital inclinations and rates.
CMEAN	I/O	C	Array of mean anomalies and rates.
CØMEG	I/O	C	Array of longitudes of ascending node and rates.
CØMEGT	I/O	C	Array of longitudes of periapsis and rates.
CSAX	I/O	C	Array of semi-major axes and rates.
DJ1900	O	C	Julian date of year 1900.
DRMAX	I/O	N/C	Maximum deviation from the reference conic before rectification.
ECEQ	O	C	Transformation matrix from Earth equatorial to ecliptic.
ENGINE	I/O	N/C	Spacecraft subsystem parameter.
EPØCH(TLNCH)	I/O	C(N)	Launch epoch.
FRCA	I/O	N/C	Specification for testing closest approach along trajectory (See Section 2.1, User's Manual).
IAUGDC	I/O	N/C	Flags specifying parameters which are used to augment the state transition matrix.
ICALL	O	C	Trajectory package initialization flag.

Variable	Input/ Output	Namelist/ Common	Definition
ICØØRD	I/O	N/C	Flag indicating relative to which body the input state corresponds.
IENRGY	I/O	N/C	Flag specifying type of power subsystem.
INIT	O	C	Cycle flag.
INTEG(IØPT(1))	O	C	Flag specifying equations to be integrated in the trajectory package.
IPRINT	I/O	N/C	Print option flags.
ISTMF	I/O	N/C	STM file flag and data cycle flag.
ISTØP	I/O	N/C	Flag specifying stopping conditions.
JPFLAG	O	C	Primary body change output flag.
KTRAJ(IØPT(2))	O	C	Control phase change output flag.
LØCS	O	C	First location in blank common available for use in the trajectory package.
MEVENT(IØPT(3))	O	C	Event detection logic flag.
MØDE	I/O	N/C	Mode specification flag.
MPLAN	O	C	Number of bodies included in the integration.
NB	I/O	N/C	Flag specifying bodies to be included in the integration.
NBØD	O	C	Number of bodies specified in NB (MPLAN-1).
NEP	I/O	N/C	Ephemeris planet designation.

Variable	Input/ Output	Namelist/ Common	Definition
XBØDY	I/Ø	N/C	Hollerith name of input body.
XPRINT	I/Ø	N/C	Trajectory print frequency (days).
ZK	I/Ø	N/C	Direction cosines of the reference star.
DUMMY	I	N	Not used.
ELVMIN	I/Ø	N/C	Minimum elevation angle.
GHZERØ	Ø	C	Greenwich hour angle at launch epoch.
IØBS	I/Ø	N/C	Index designating location of astronomical observatory in STALØC.
KARDS	I/Ø	N/C	Number of formatted print schedule cards to be read during a REFSEP run.
PRNML	I	N	Logical flag specifying that the $TRAJ namelist be printed (TRUE) or not be printed (FALSE).
STALØC	I/Ø	N/C	Tracking station coordinates.

Local Variables:

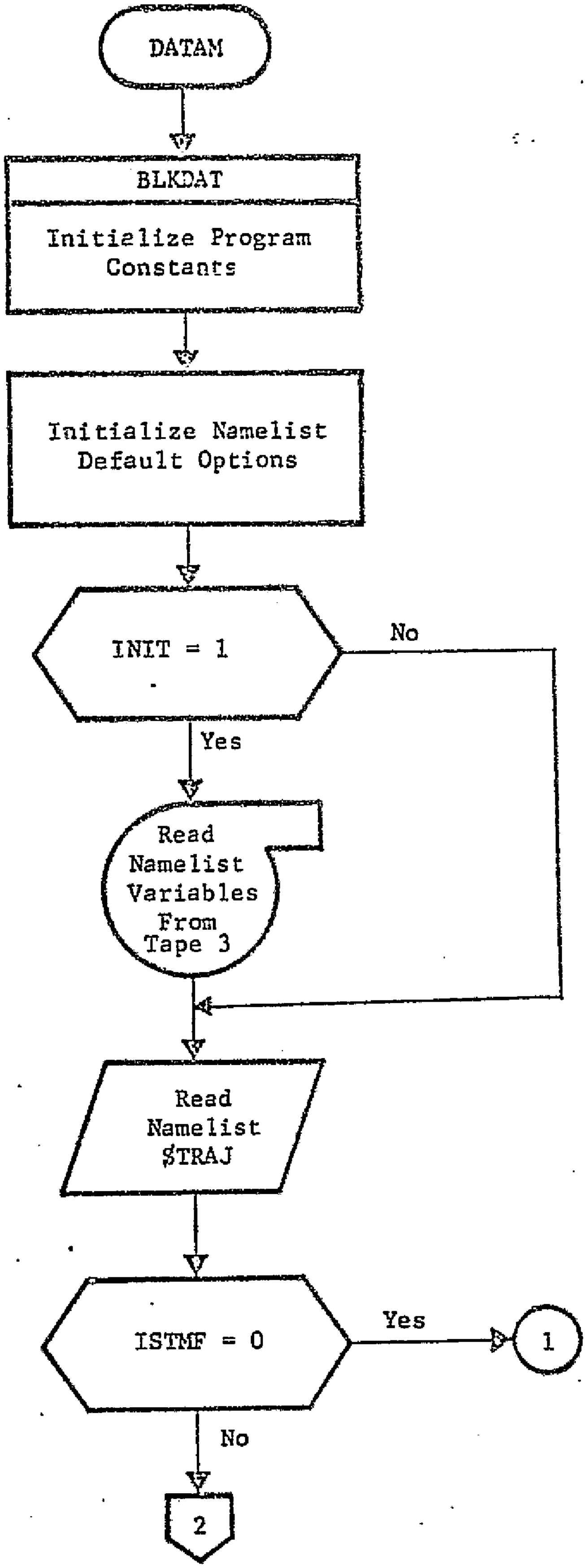
Variable	Definition
AO, A1, A2, A3	Constants used in the obliquity computations.
DJCENT	Days in a Julian Century.
D10K	Constant 10^4.
IØPT	Option flags used to set parameters in TRAJ.
JMAX	Number of thrust control phases.
STATER	Magnitude of initial position vector.
STATEV	Magnitude of initial velocity vector.

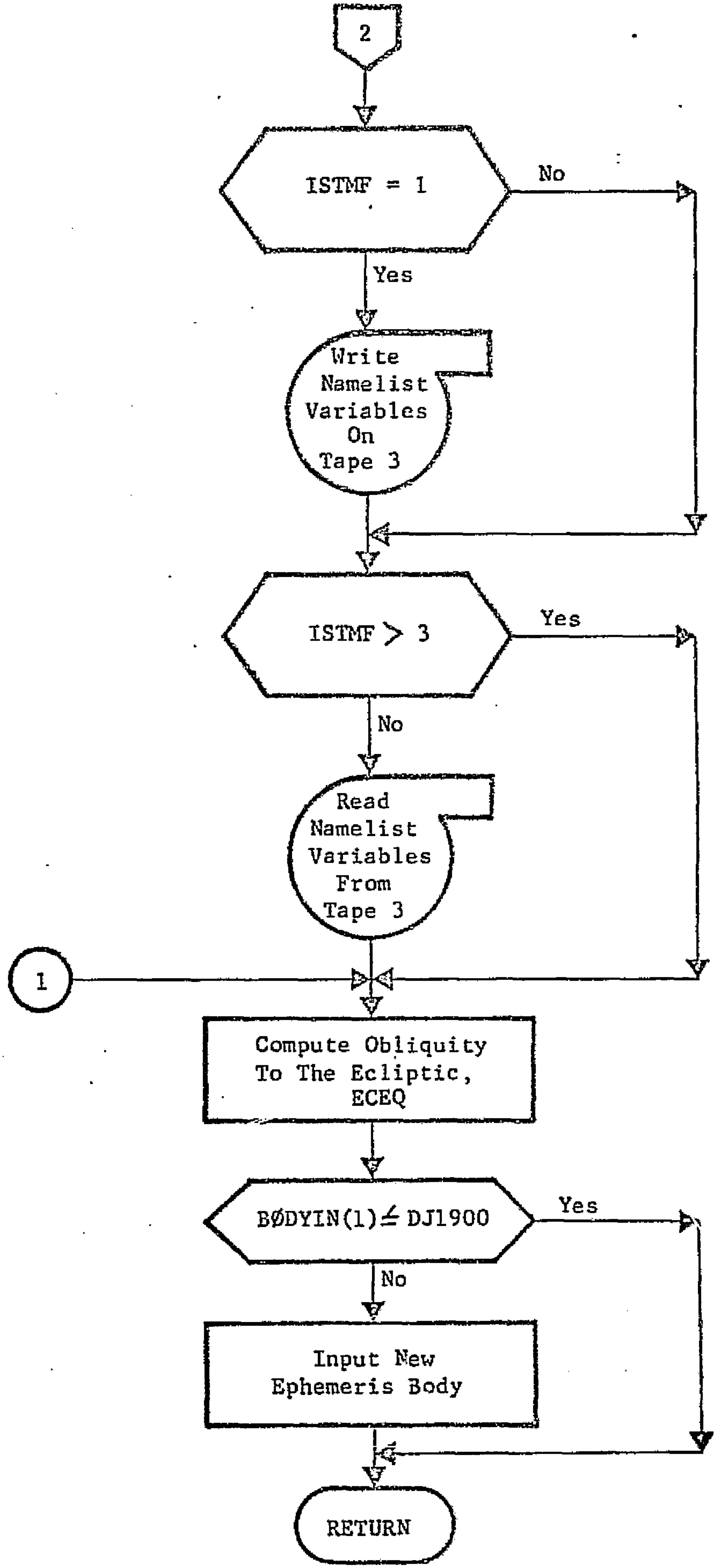
Subroutines Called: BLKDAT, ZERØM, MMAB, VECMAG, TIME

Calling Subroutine: MAPSEP

Common Blocks: CØNST, EDIT, EPHEM, TIME, TRAJ1, TRAJ2, WØRK, TRKDAT

PRECEDING PAGE BLANK NOT FILMED


DATAM
BLKDAT
Initialize Program Constants
Initialize Namelist Default Options
INIT = 1
No
Yes
Read Namelist Variables From Tape 3
Read Namelist $TRAJ
ISTMF = 0
Yes
1
No
2


2
ISTMF = 1
No
Yes
Write Namelist Variables On Tape 3
ISTMF > 3
Yes
No
Read Namelist Variables From Tape 3
1
Compute Obliquity To The Ecliptic, ECEQ
BØDYIN(1) ≤ DJ1900
Yes
No
Input New Ephemeris Body
RETURN

3.1.3 Subroutine: TIME (DAY, IYR, MØ, IDAY, IHR, MIN, SEC, ICØDE)

Purpose: TIME converts a Julian Date to the corresponding calendar date or a calendar date to the corresponding Julian Date.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
DAY	I/O	A	Julian Date.
IYR	I/O	A	Calendar year.
MØ	I/O	A	Month.
IDAY	I/O	A	Day.
IHR	I/O	A	Hour.
MIN	I/O	A	Minute.
SEC	I/O	A	Second.
ICØDE	I	A	Flag that determines whether to convert from a Julian Date to calendar day or vice versa. = 0, Convert to a Julian Date ≠ 0, Convert from a Julian Date

Subroutines Called: None

Calling Subroutine: DATAM

Common Blocks: None

3.2 Subroutine: TØPSEP

Purpose: To execute the proper submode operation.

Remarks: TØPSEP is the primary overlay which controls the targeting and optimization mode.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
IMØDE	I	C	Submode designation.
MØDE	I	C	Mode designation. -1, Cycle back within mode 1, Cycle back to MAPSEP main

Local Variables:

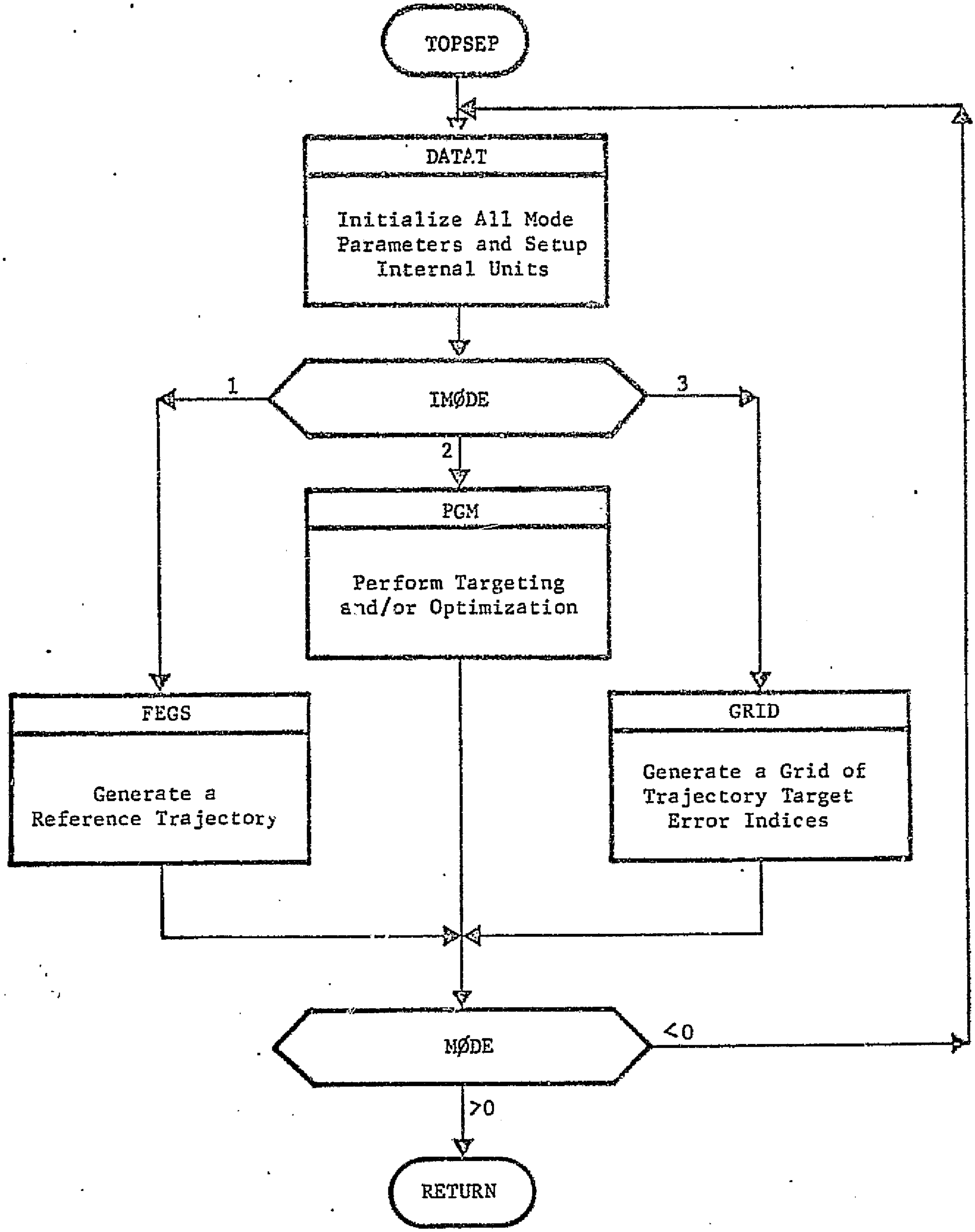
Variable	Definition
WØRK	Working storage.

Subroutines Called: DATAT, FEGS, GRID, PGM

Calling Subroutines: MAPSEP

Common Blocks: (BLANK), ALTFIL, CØNST, EDIT, EPHEM, GRID, TIME, TØP1, TØP2, TRAJ1, TRAJ2, WØRK

Logic Flow:


TOPSEP
DATAT
Initialize All Mode Parameters and Setup Internal Units
IMØDE
1
2
3
PGM
Perform Targeting and/or Optimization
FEGS
Generate a Reference Trajectory
GRID
Generate a Grid of Trajectory Target Error Indices
MØDE
<0
>0
RETURN

3.2.1 Subroutine: BUCKET (X, Y, N, XX, YY, NP)

Purpose: To sort a set of independent elements in ascending order and to find a right bounded minimum from the associated set of dependent elements.

Remarks: This routine is used in preparation for the polynomial curve fitting routine, MINMUM, to aid in calculating trial control profiles. BUCKET sorts pairs of elements (X_i, Y_i) in ascending order of the elements X_i to form the pairs of elements (XX_i, YY_i) and locates the element YY_{NP} such that

$$YY_{NP} < YY_{NP+1}$$

If this condition cannot be satisfied the pointer, NP, is set to zero to indicate that no right bounded minimum exists.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
N	I	A	Number of elements to be sorted.
NP	Ø	A	Pointer to a minimum dependent element.
X	I	A	Vector of independent elements to be sorted.
XX	Ø	A	Vector of ordered independent elements.

Variable	Input/ Output	Agrument/ Common	Definition
Y	I	A	Vector of dependent elements associated with X.
YY	Ø	A	Vector of dependent elements associated with XX.

Local Variables:

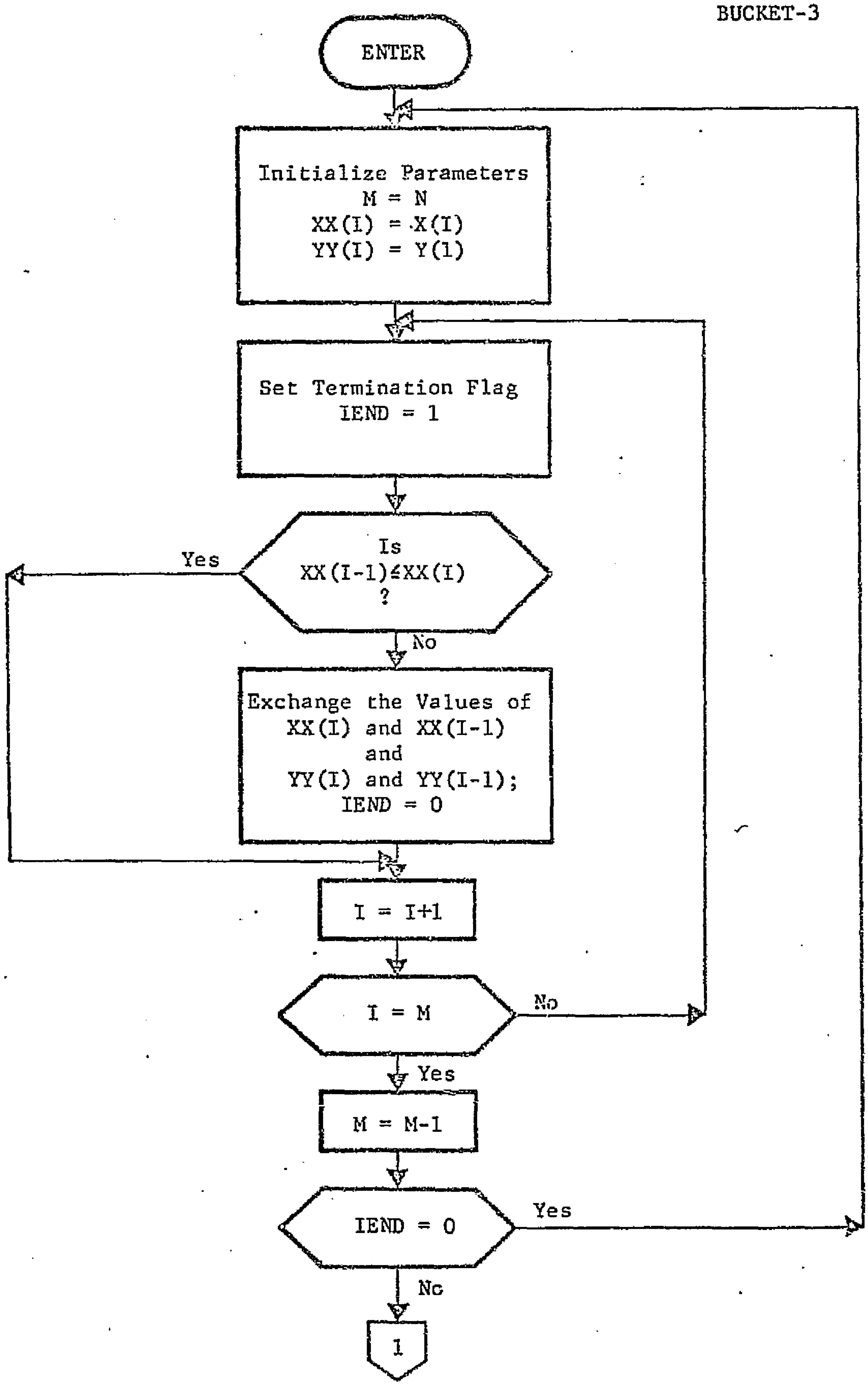
Variable	Definition
IEND	Termination flag.
SAVE	Intermediate variable.

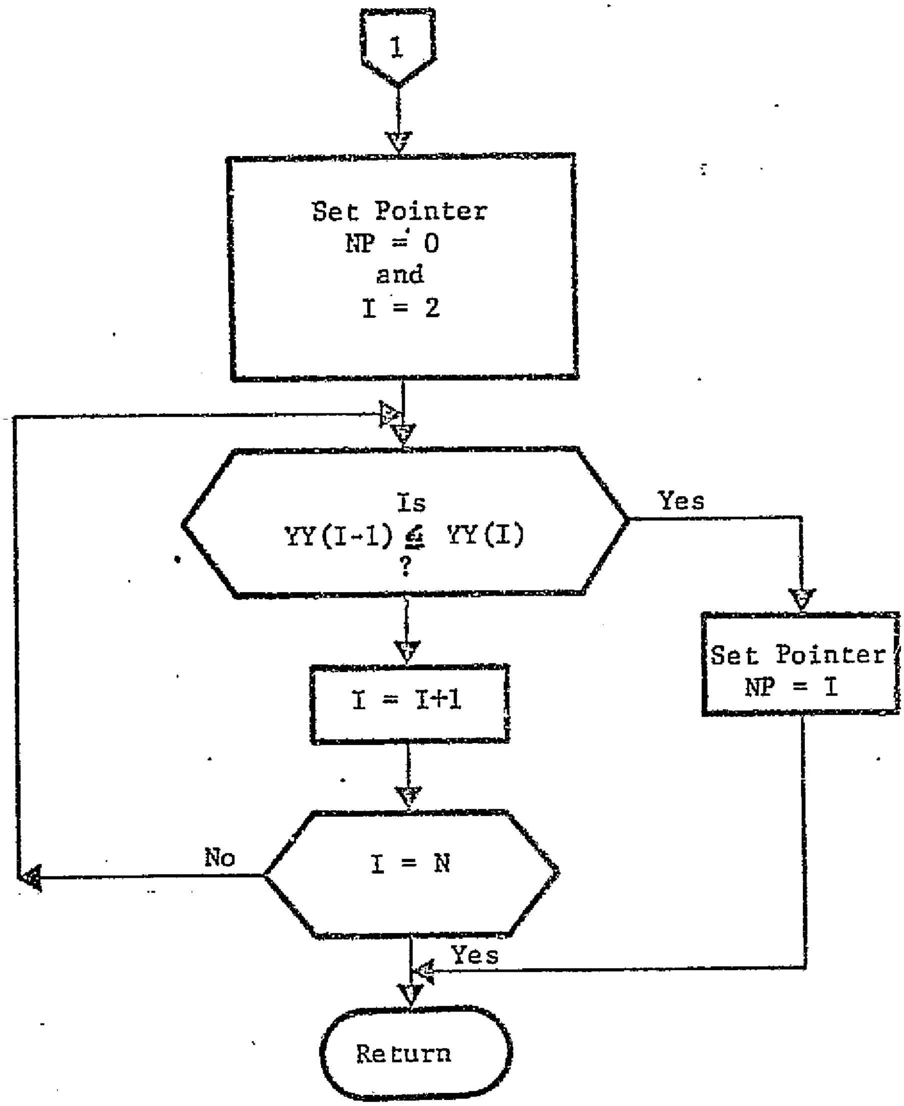
Subroutines Called: None

Calling Subroutines: GENMIN

Common Blocks: None

BUCKET-3


ENTER
Initialize Parameters
M = N
XX(I) = X(I)
YY(I) = Y(1)
Set Termination Flag
IEND = 1
Is
XX(I-1)≤XX(I)
?
Yes
No
Exchange the Values of
XX(I) and XX(I-1)
and
YY(I) and YY(I-1);
IEND = 0
I = I+1
I = M
No
Yes
M = M-1
IEND = 0
Yes
No
1


1
Set Pointer
NP = 0
and
I = 2
Is
YY(I-1) ≤ YY(I)
?
Yes
Set Pointer
NP = I
I = I+1
No
I = N
Yes
Return

3.2.2 Subroutine: DATAT

Purpose: To read input data and initialize the trajectory targeting and optimization mode.

Method: After DATAT executes the default value initialization, the namelist $TØPSEP is read. The dimensions and definitions for variables contained in this namelist are discussed in detail in the TØPSEP section of the User's Manual. The input data are processed and stored in labeled common for subsequent use in any of the three possible submodes. User options specified by input determine the degree of data preparation and the logic operations within the main cycle of the program.

Remarks: Some variables appearing in DATAT are initialized from the namelist with units specified in the User's Manual. Before they are transmitted to other routines, they are converted, if necessary, to internal operational units which are: kg, kw, km, sec, km/sec, and radians.

Input/Output:

Variable	Input/ Output	Namelist/ Common	Definition
BIG	I	C	Large constant, 1.E20
BTØL	I	N/C	Tolerance on control bounds.

Variable	Input/ Output	Namelist/ Common	Definition
CHI	O	C	In plane ΔV direction angle at injection.
CNTRØL	O	C	Initial values of all possible controls other than thrust controls.
CNVRTT	O	C	Conversion constants from input units to internal units for selected targets.
CNVRTU	O	C	Conversion constants from input units to internal units for selected controls.
DELVO	O	C	Injection $\|\Delta \underline{V}\|$.
DFMAX	I/O	N/C	Maximum increase allowed in the cost index (F) per iteration.
DP2	I/O	N/C	Estimated region of linearity in the control space.
E	O	C	Target errors of the current trajectory.
ENGINE(1)	I	N/C	Power from solar panels at 1 A.U.
ENGINE(10)	I	N/C	S/C exhaust velocity.
EPSØN	I	N/C	Scalar multiple for control perturbations.
ETLØUT	O	C	Target tolerances in print-out units.
ETØL	O	C	Target tolerances.
G	I/O	N/C	Performance gradient.
GØUT	O	C	Performance gradient in print-out units.
GTRIAL	I/O	N/C	One-dimensional search constants.

Variable	Input/ Output	Namelist/ Common	Definition
H	I/O	N/C	Control perturbation array.
HMULT	I/O	N/C	Vector of scalar multiples of the H array to determine the second step of all controls in the control grid.
HØUT	O	C	Control perturbation array in print-out units.
ICYCLE	I/O	C	Mode cycle flag.
IMØDE	I/O	N/C	TOPSEP submode designation.
INACTV	O	C	Vector denoting which controls are active, or bounds, or within bound tolerance regions.
INJLØC	O	C	Index of the control preceding the injection controls in U.
INSG	I/O	N/C	Flag set when S and G are input through namelist.
ITERAT	O	C	Iteration counter.
IWATE	I/O	N/C	Flag designating the desired control weighting schemes.
JMAX	O	C	Number of mission thrust phases.
JWATE	I/O	N/C	Flag designating target weighting.
KMAX	O	C	Number of thrust controls (THRUST(I,J)) chosen to be elements in U.
KØNVRJ	O	C	Convergence flag.
LABEL	O	C	Hollerith names of all possible targets.

Variable	Input/ Output	Namelist/ Common	Definition
LABELT	O	C	Hollerith names of chosen targets.
LØCCDC	O	C	Blank common storage location for the inner products of the weighted sensitivity matrix columns.
LØCCM	O	C	Blank common location for storage of the magnitude of the weighted sensitivity column vectors.
LØCDU	O	C	Blank common location of the total control correction vector (not scaled by GAMA).
LØCDU1	O	C	Blank common location of the performance control correction vector (not scaled by GAMA).
LØCDU2	O	C	Blank common location of the constraint control correction vector (not scaled by GAMA).
LØCE1	O	C	Blank common location of the target errors associated with the first step of the control grid.
LØCE2	O	C	Blank common location of the target errors associated with the second step of the control grid.
LØCEM1	O	C	Blank common location of the target error indices associated with the first step of the control grid.
LØCEM2	O	C	Blank common location of the target error indices associated with the second step of the control grid.

Variable	Input/ Output	Namelist/ Common	Definition
LØCEN	O	C	Blank common location of the nominal trajectory target errors in the grid mode.
LØCF1	O	C	Blank common location of the performance indices associated with the first step of the control grid.
LØCF2	O	C	Blank common location of the performance indices associated with the second step of the control grid.
LØCRFM	O	C	Blank common location of the S/C masses evaluated at event times for the reference and all trial trajectories in a single iteration.
LØCSDU	O	C	Blank common storage location for the original control correction vectors when a number of controls must be dropped during an iteration.
LØCSI*	O	C	Blank common location of the pseudo inverse of the weighted sensitivity matrix.
LØCSWG	O	C	Blank common storage location for the original weighted performance gradient when a number of controls must be dropped during an iteration.
LØCSWS	O	C	Blank common storage location for the original weighted sensitivity matrix when a number of controls must be dropped during an iteration.

*May be in compressed form if controls have been dropped during the iteration.

Variable	Input/ Output	Namelist/ Common	Definition
LØCTS	O	C	Blank common location of event times for the reference and all trial trajectories in a single iteration.
LØCUL	O	C	Blank common location of minimum and maximum control bounds.
LØCWG*	O	C	Blank common location of the weighted performance gradient.
LØCWS*	O	C	Blank common location of the weighted sensitivity matrix.
LØCWU	O	C	Blank common location of the control weights.
LØCXR	O	C	Blank common location of the 6-component state vectors associated with the event times of the reference and all the trial trajectories of a single iteration.
MPRINT	I/O	N/C	Flag designating TOPSEP print options.
NLP	I	C	Integer designation for launch planet.
NMAX	I/O	N/C	Maximum number of iterations.
NT	O	C	Number of targets.
NTNP	O	C	Vector of primary bodies associated with the event times of the reference and all trial trajectories in a single iteration.

*May be in compressed form if controls have been dropped during the iteration.

Variable	Input/ Output	Namelist/ Common	Definition
NTPH	O	C	Vector of control phase numbers associated with the event times of the reference and all trial trajectories in a single iteration.
NTR	O	C	Trial trajectory counter (NTR = 1 indicates the iteration reference trajectory).
NTYPE	O	C	Flag designating the type of control correction to be made during an iteration.
NU	O	C	Number of controls.
ØPTEND	I/O	N/C	Cosine of the optimization angle which is used to test convergence in the targeting and optimization mode.
ØSCALE	I/O	N/C	Scale on the performance index when simultaneously targeting and optimizing.
PCT	I/O	N/C	Percentage of the target error to be removed during an iteration.
PRTURB	O	C	Vector of control perturbations; summary of H array.
PSI	O	C	Out of plane ΔV direction angle at injection.
P1	O	C	Vector of net cost values for the reference and all trial trajectories evaluated during a single iteration.
P1P2	O	C	Vector of combined target error indices and net cost, values for the reference and all trail trajectories evaluated during a single iteration.

Variable	Input/ Output	Namelist/ Common	Definition
P2	O	C	Vector of target error indices for the reference and all trial trajectories evaluated during a single iteration.
RAD	I	C	Number of degrees in one radian.
S	I/O	N/C	Target sensitivity matrix.
SCMASS	I	C	S/C initial mass.
SØUT	O	C	Target sensitivity matrix in print-out units.
STATEO	I	C	Initial state.
STØL	I	N/C	Test variable for determining linearly dependent columns of the weighted sensitivity matrix.
STØRE	I/O	C	Blank common variable.
TARGET	I/O	N/C	Vector of desired target values.
TARØUT	O	C	Desired target values in print-out units.
TARTØL	I/O	N/C	Vector of all possible target tolerances.
THRUST	I	C	Mission thrust controls.
TLØW	I	N/C	Limit of target error index below which optimization only is performed.
TM	I	C	Number of seconds in a day.
TSTART	I	C	Reference trajectory start time.
TUP	I	N/C	Limit of target error index above which simultaneous targeting and optimization is discontinued and targeting

Variable	Input/ Output	Namelist/ Common	Definition
U	Ø	C	Selection of controls for the specified mode run.
ULIMIT	I	N	Control bounds.
UWATE	I/Ø	N/C	User input weights on controls.
VPARK	Ø	C	Parking orbit velocity at injection.
WE	Ø	C	Vector of target weights.
XMM	Ø	C	Mean motion of s/c in parking orbit.
AZMAX	I/Ø	N/C	Maximum launch azimuth constraint.
AZMIN	I/Ø	N/C	Minimum launch azimuth constraint.
IASTM	I/Ø	N/C	Flag specifying the method of computing the targeting sensitivity matrix.
PRNML	I	N	Logical flag specifying that the namelist $TRAJ be printed (TRUE) or not be printed (FALSE).
RP1	I/Ø	N/C	Inner parking orbit radius.
TGFUEL	I/Ø	N/C	Fuel capacity of tug.
TUGISP	I/Ø	N C	Specific impulse of tug.
TUGWT	I/Ø	N/C	Dry weight of tug.
TUG	Ø	C	Logical flag designating injection computations.

Local Variables:

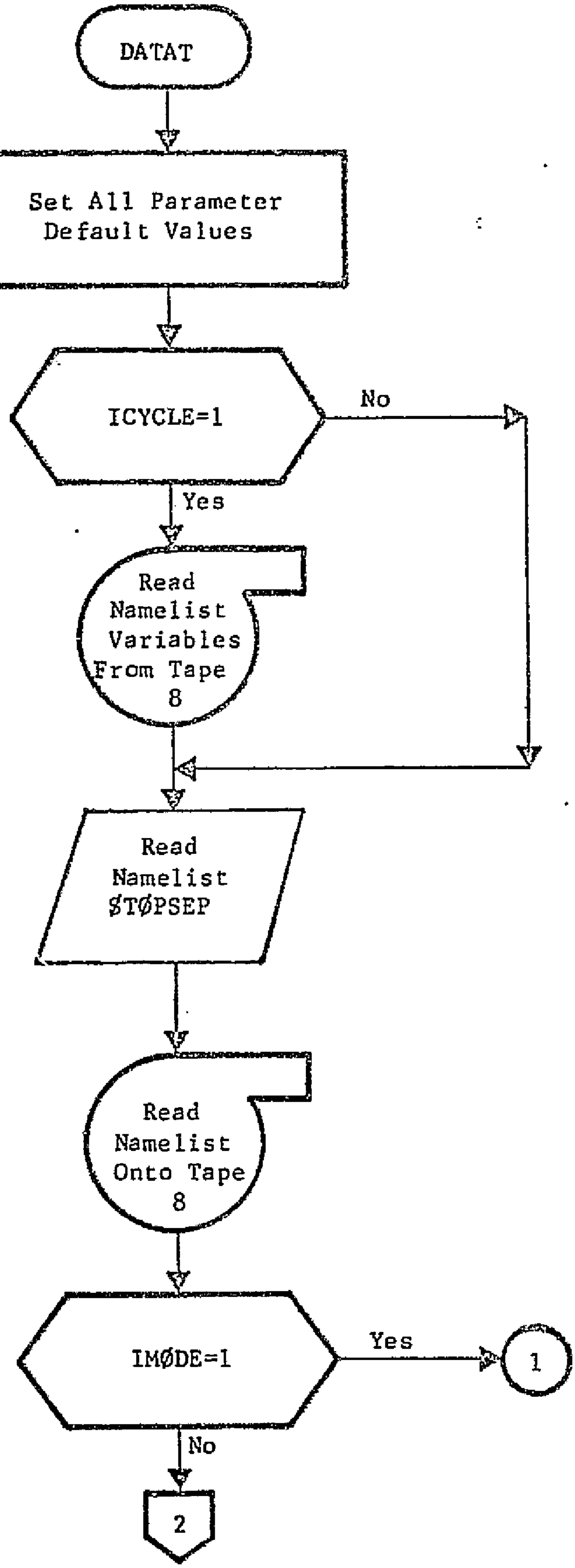
Variable	Definition
KØUNT	Control counter.
TIME	Mission time corresponding to the implementation of controls chosen from the elements of the THRUST array.

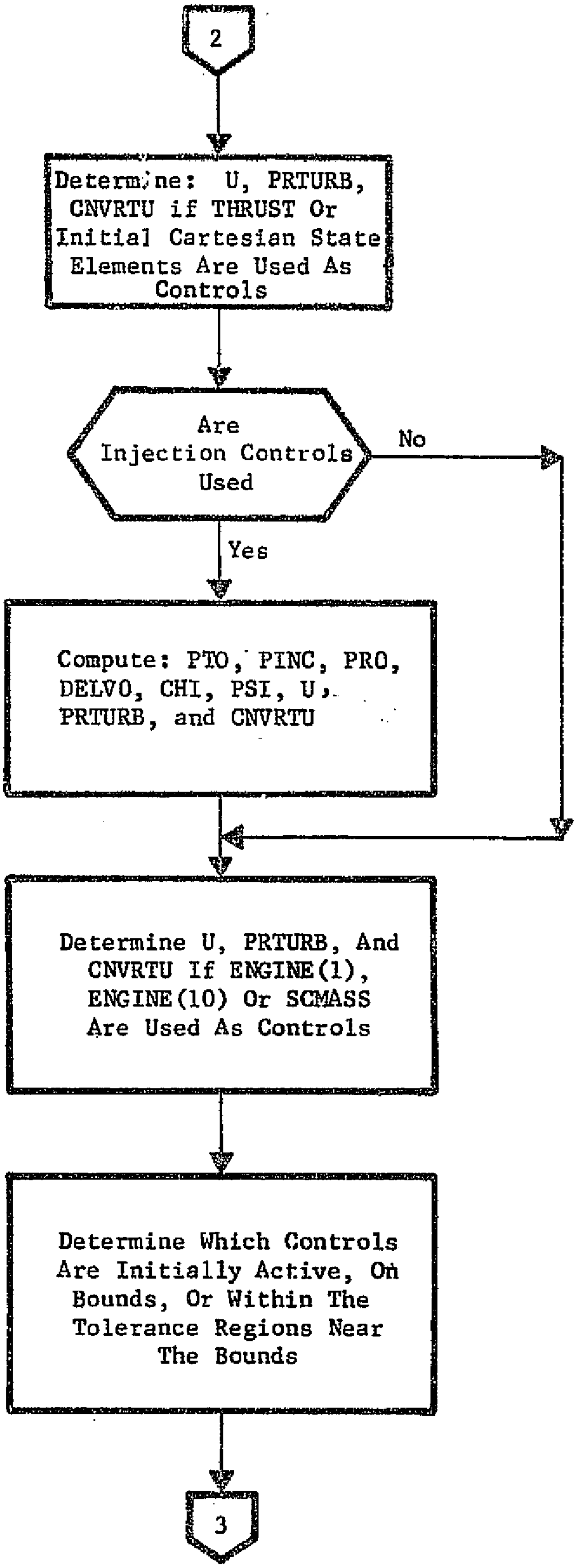
<u>Subroutines Called</u>: ZEROM, CØPY, UXV, UNITV, SCALE, SUB, VECMAG, UDØTV, PRINTD, INJECT

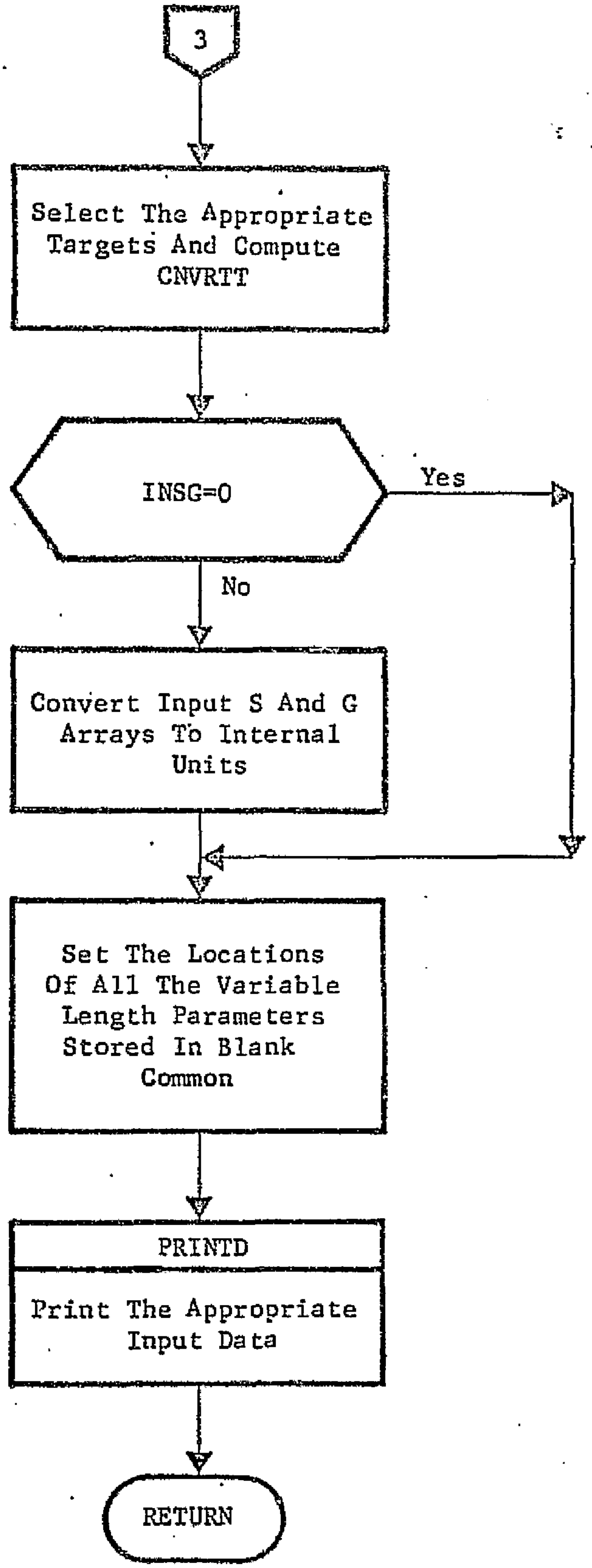
<u>Calling Subroutines</u>: TØPSEP

<u>Common Blocks</u>: (BLANK), CØNST, CYCLE, EDIT, EPHEM, GRID, PRINT, PRINTH, TIME, TØP1, TØP2, TRAJ1, TRAJ2, WØRK, IASTM, TUG

Logic Flow:


DATAT
Set All Parameter Default Values
ICYCLE=1
No
Yes
Read Namelist Variables From Tape 8
Read Namelist $TØPSEP
Read Namelist Onto Tape 8
IMØDE=1
Yes
1
No
2


2
Determine: U, PRTURB, CNVRTU if THRUST Or Initial Cartesian State Elements Are Used As Controls
Are Injection Controls Used
No
Yes
Compute: PTO, PINC, PRO, DELVO, CHI, PSI, U, PRTURB, and CNVRTU
Determine U, PRTURB, And CNVRTU If ENGINE(1), ENGINE(10) Or SCMASS Are Used As Controls
Determine Which Controls Are Initially Active, On Bounds, Or Within The Tolerance Regions Near The Bounds
3


3
Select The Appropriate Targets And Compute CNVRTT
INSG=0
Yes
No
Convert Input S And G Arrays To Internal Units
Set The Locations Of All The Variable Length Parameters Stored In Blank Common
PRINTD
Print The Appropriate Input Data
RETURN

3.2.3 Subroutine: DELU (WS, WG, DPSI, DP2, NT, NU, NTYPE, SINV, PG2, DU1, DU2, DU).

Purpose: To compute the control correction based upon the method of projected gradients.

Method: The projected gradient algorithm used in TOPSEP is described as follows. Let:

$\underline{U}$ = Set of control parameters;

$\underline{E}$ = Set of target errors;

F = Performance index;

G = Performance gradient $\left(\frac{\partial F}{\partial \underline{U}}\right)$;

T = Set of targets;

S = Sensitivity matrix $\left(\frac{\partial T}{\partial \underline{U}}\right)$;

We seek a control correction $\Delta \underline{U}$ to increase the performance (decrease the cost) and decrease the target error. Then

$$\Delta \underline{U} = \alpha \Delta \underline{U}_1 + \beta \Delta \underline{U}_2$$

where

$$\Delta \underline{U}_2 = -S^T (SS^T)^{-1} \underline{E}$$

$$\Delta \underline{U}_1 = \frac{-\sqrt{\Delta \underline{U}_2^T \quad \Delta \underline{U}_2}\ (I-P)\ \underline{G}}{\| (I-P)\ \underline{G} \|}$$

and

$$P = S^T (SS^T)^{-1} S$$

$$\alpha = \begin{cases} 0, & \text{for targeting only} \\ 1, & \text{for optimization} \end{cases}$$

$$\beta = \begin{cases} 0, & \text{for optimization only} \\ 1, & \text{for targeting} \end{cases}$$

Remarks: DELU is called only after transforming the control space to a weighted space. Thus, WS and WG are a weighted target sensitivity matrix and a weighted performance gradient respectively. The control corrections, therefore, are also weighted.

The performance correction is modified to account for an estimated region of linearity (DP2). This control correction may then be represented as follows:

$$\Delta U_1 = \text{REGIØN} * (I-P)\,G$$

$$\text{REGIØN} = -\sqrt{\frac{E^T (SS^T)^{-1} E * (1+DP2^2)}{G^T G - (SG)^T (SS^T)^{-1} (SG)}}$$

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
DPSI	I	A	Target error to be removed during current iteration.
DP2	I/O	A	Estimated region of linearity in the control space.

Variable	Input/ Output	Argument/ Common	Definition
DU	O	A	Total control correction vector (not scaled).
DU1	O	A	Performance control vector (not scaled).
DU2	O	A	Constraint control correction (not scaled).
NT	I	A	Number of controls.
NTYPE	I	A	Flag designating the type of control correction to be made during the current iteration.
NU	I	A	Number of controls.
PG2	O	A	Magnitude of the projected gradient squared.
SINV	O	A	Pseudo-inverse of the target sensitivity matrix if NU NT; actual inverse of target sensitivity matrix if NU = NT.
WG	I	A	Performance gradient.
WS	I	A	Target sensitivity matrix.

ALPHA

Local Variables:

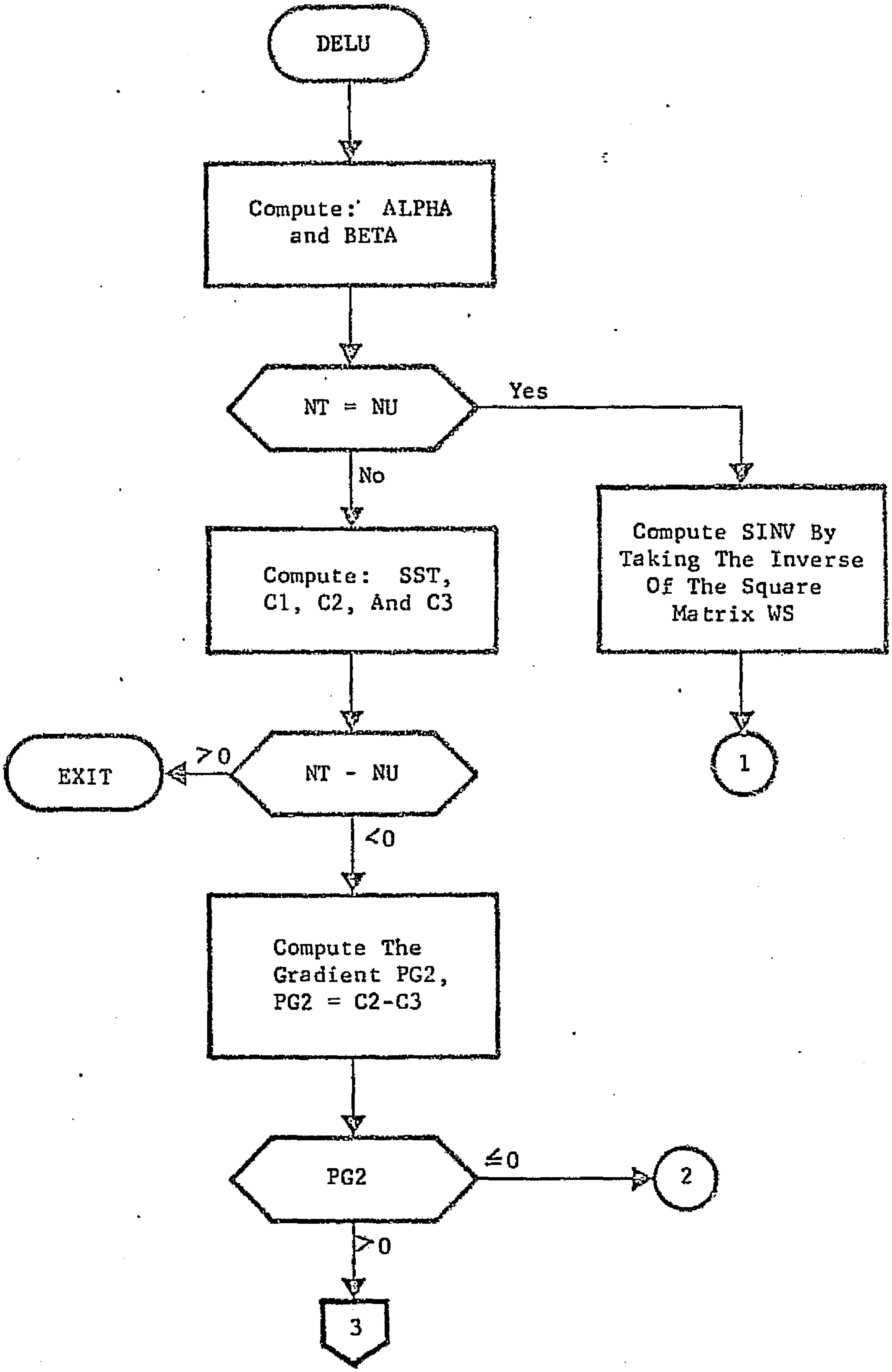
Variable	Definition
ALPHA	Scale on DU1 when computing DU; if not making a performance correction ALPHA set to 0, otherwise set to 1.
BETA	Scale on DU2 when computing DU; if not making a constraint correction BETA set to 0, otherwise set to 1.
CI	$E^T * (S*S^T)^{-1} * E$

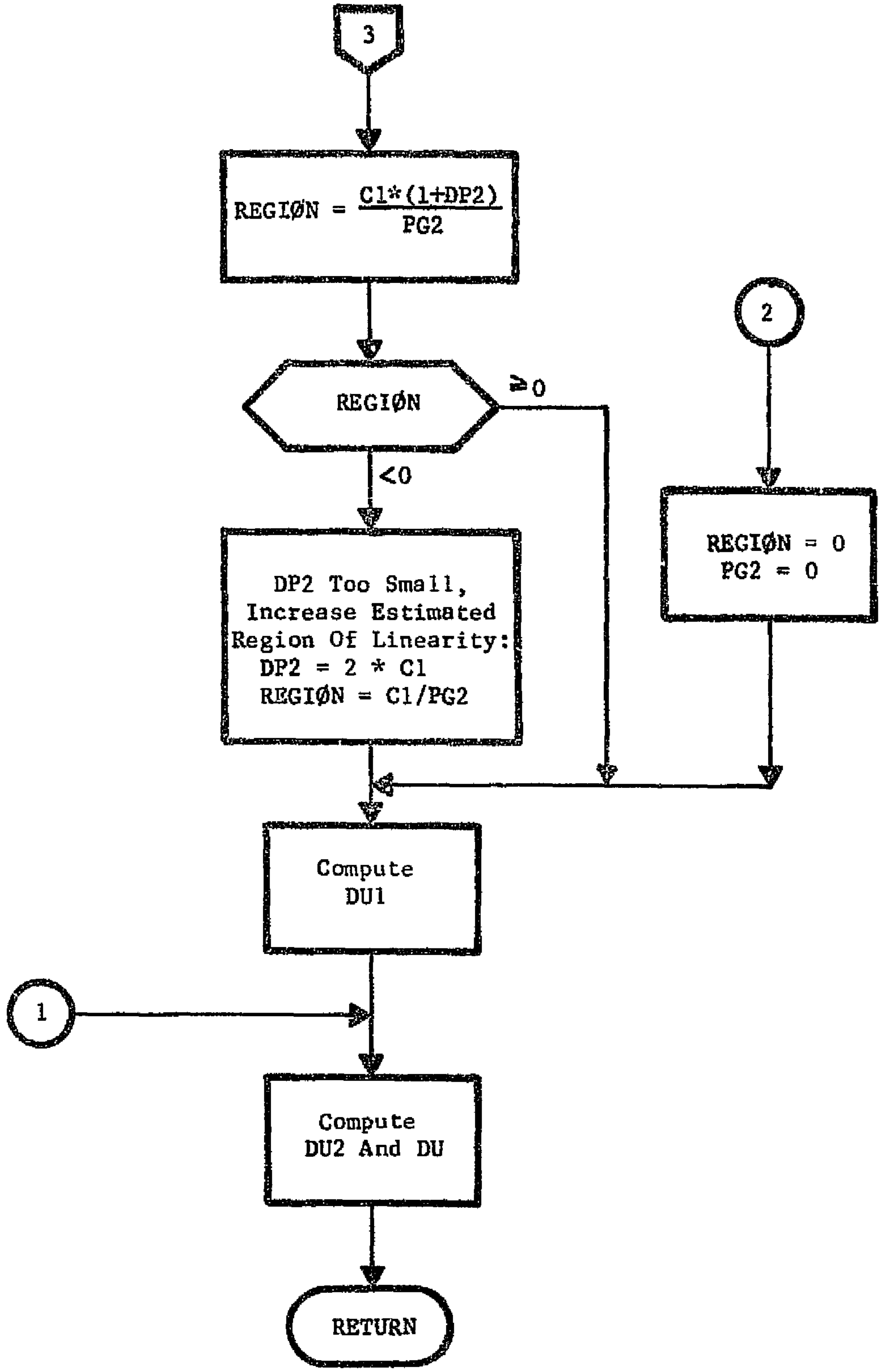
Variable	Definition
C2	$G^T * G$
C3	$(S*G)^T * (S*S^T)^{-1} * (S*G)$
P (=WØRK (43))	$S^T * (S*S^T)^{-1} * S*G$
REGIØN	Scale on performance correction accounting for the assumed region of linearity.
SG (=WØRK (37))	S*G
SST (=WØRK (1))	$S*S^T$

Subroutines Called: CØPY, INVSQM, MMAB, MMABT, MMATB, MMATBA, ZERØM

Calling Subroutines: SIZE

Common Blocks: EDIT, WØRK


DELU
Compute: ALPHA and BETA
NT = NU
Yes
No
Compute SINV By Taking The Inverse Of The Square Matrix WS
1
Compute: SST, C1, C2, And C3
EXIT
>0
NT - NU
<0
Compute The Gradient PG2, PG2 = C2-C3
PG2
≤0
2
>0
3


3
REGIØN = C1*(1+DP2) / PG2
REGIØN
≥0
<0
2
DP2 Too Small, Increase Estimated Region Of Linearity:
DP2 = 2 * C1
REGIØN = C1/PG2
REGIØN = 0
PG2 = 0
Compute DU1
1
Compute DU2 And DU
RETURN

3.2.3A Subroutine: DIRECT (DU1, DU2, DU, SINV, ULIMIT, WG, WS, WU, NUD, NTD)

Purpose: To compute the control correction, Δu.

Method: The method of projected gradients is used to compute Δu. Preliminary computations include:

- Determining linear dependency among columns of the sensitivity matrix, S, thus averting numerical problems when computing the pseudo-inverse of S.
- Determining which controls lie on their respective bounds, if any, and which control corrections violate the control constraints.
- Determining the maximum allowable scale factor for the current iteration.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
BIG	I	C	Large constant, 1.E20.
CTHETA	O	C	Cosine of optimization angle.
DFMAX	I	C	Maximum increase allowed in the cost index (F) per iteration.
DPSI	O	C	Target error to be removed during current iteration.
DP2	I/O	C	Estimated region of linearity in the control space.

Variable	Input/ Output	Argument/ Common	Definition
DU	O	A	Unscaled total control correction.
DU1	O	A	Unscaled performance control correction.
DU2	O	A	Unscaled constraint control correction.
E	O	C	Target errors of the current trajectory.
EMAG	O	C	Target error index.
G	O	C	Performance gradient.
GAMA	O	C	Scale factor providing the best control change.
GAMMA	O	C	Vector of trial trajectory control change scale factors.
GTRIAL	I/O	C	One-dimensional search constants.
INACTV	I/O	C	Vector denoting which controls are active (1), on bounds (0), or within bound tol.
KGMAX	O	C	Index identifying the control which will reach bound if $\Delta \underline{u}$ is scaled by GMAX.
ITERAT	I	C	Iteration counter.
LØCCDC	I	C	Blank common location of the inner products of the columns of the sensitivity matrix.
LØCCM	I	C	Blank common location of the magnitude of the sensitivity column vectors.
LØCSDU	I	C	Blank common storage location for the original control correction vectors when a number of controls must be dropped during an iteration.

Variable	Input/ Output	Argument/ Common	Definition
LØCSWG	I	C	Blank common storage location for the original weighted performance gradient when a number of controls must be dropped during an iteration.
LØCSWS	I	C	Blank common storage location for the original weighted sensitivity matrix when a number of controls must be dropped during an iteration.
MPRINT	I	C	Array of TOPSEP print flags.
NT	I	C	Number of targets.
NTD	I	A	Integer used to variably dimension SINV and WS.
NTYPE	I	C	Flag designating the type of control correction to be made during an iteration.
NU	I	C	Number of controls.
NUD	I	A	Integer used to variably dimension DU, DU1, DU2, SINV, ULIMIT, WG, WS and WU.
ØSCALE	I	C	Scale on the cost index when simultaneously targeting and optimizing.
PCT	I	C	Percentage of the target error to be removed during an iteration.
P1	O	C	Vector of net cost values for the reference and all trial trajectories evaluated during a single iteration.

Variable	Input/ Output	Argument/ Common	Definition
P1P2	O	C	Vector of combined target error indices and net cost values for the reference and all trial trajectories evaluated during a single iteration.
P2	O	C	Vector of target error indices for the reference and all trial trajectories evaluated during a single iteration.
S	I	C	Target sensitivity matrix.
SINV	O	A	Test variable for determining linearly dependent columns of the weighted sensitivity matrix.
U	I	C	Selection of controls.
ULIMIT	I	A	Bounds on controls.
WE	I	C	Vector of target weights.
WG	O	A	Weighted performance gradient.
WS	O	A	Weighted sensitivity matrix.
WU	O	A	Control weights.
DP1DS	O	C	The first derivative of the net cost function (P1) evaluated at $\gamma = 0$.
DP12DS	O	C	The first derivative of the combined net cost function and target error function (P1P2) evaluated at $\gamma = 0$.
DP2DS	O	C	The first derivative of the target error function (P2) evaluated at $\gamma = 0$.

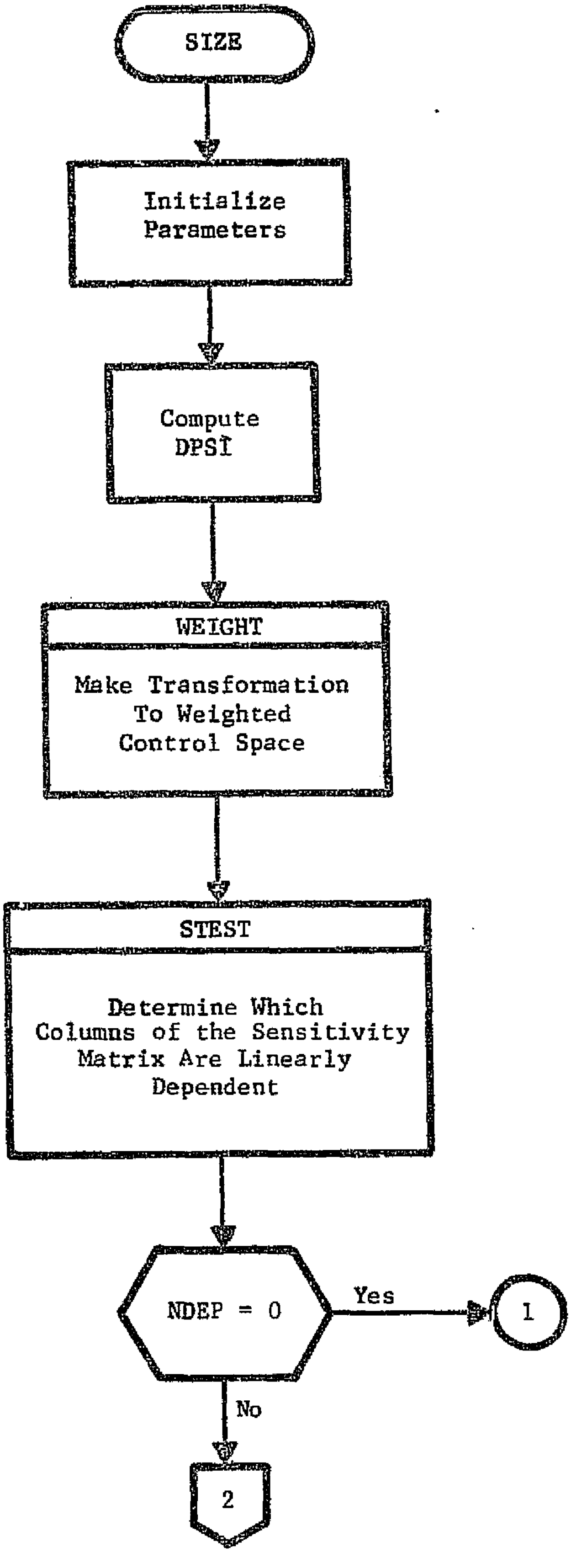
Local Variables:

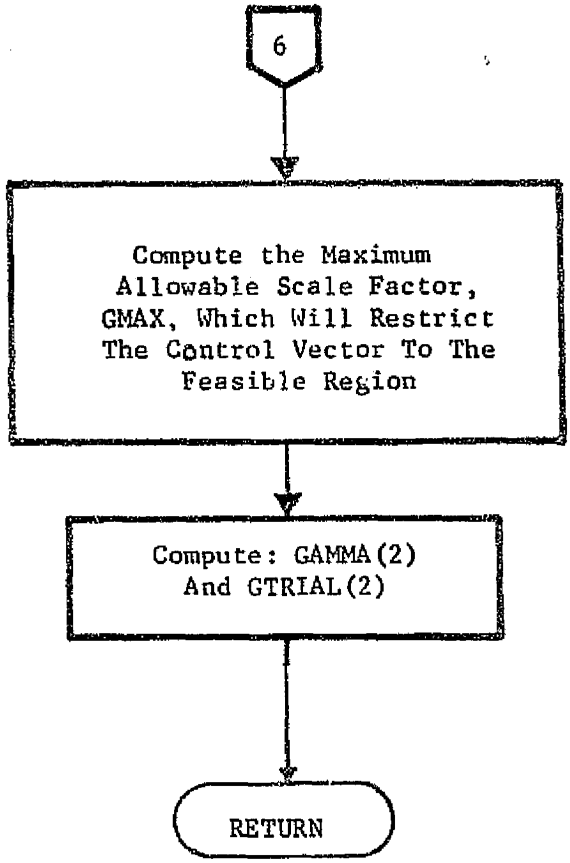
Variable	Definition
DU1MAG (=WORK(1))	Magnitude of $\Delta \underline{u}_1$.
EPRIME (=WORK(10))	Weighted target errors.
ES (=WORK (16))	$\underline{E}^T S$.
GAM (=WORK (36))	Vector of maximum allowable scale factors for each element of the control correction.
GFMAX	Estimate of the scale factor which will cause the DFMAX constraint to be violated.
KDEP	Number of controls on bounds.
LDEP	Vector indicating which controls are to be dropped from the control correction.
MU	Number of active controls in the current iteration.
SSINV (=WORK (80))	Storage for the pseudo-inverse of the sensitivity matrix.
UNEW (=WORK (60))	Updated control vector used to compute INACTV.

Subroutines Called: CØPY, GENMIN, STEP, DELTU

Calling Subroutines: PGM

Common Blocks: (BLANK), CØNST, EDIT, TØP1, TØP2, WØRK, SIZE


SIZE
Initialize Parameters
Compute DPSI
WEIGHT
Make Transformation To Weighted Control Space
STEST
Determine Which Columns of the Sensitivity Matrix Are Linearly Dependent
NDEP = 0
Yes
1
No
2


6
Compute the Maximum Allowable Scale Factor, GMAX, Which Will Restrict The Control Vector To The Feasible Region
Compute: GAMMA(2) And GTRIAL(2)
RETURN

3.2.3B Subroutine: DTDUO

Purpose: To compute the appropriate columns of the targeting sensitivity matrix which relate changes in target values to changes in the initial state.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
ETA	I	A	Sensitivity of targets to changes in final state
IJH	I	C	Array indicating active controls
M	I	A	Number of targets
N	I	A	Number of controls
PHI	I	C	State transition matrix
SPRIME	Ø	A	Partition of sensitivity matrix

Local Variables

Variable	Definition
DXFDXO	Sensitivity of final state to changes in selected elements of initial state

Subroutines Called: CØPY, MMAB

Calling Subroutines: STMTAR

Common Blocks: IASTM, TØP2, WØRK

Logic Flow: See listing

3.2.4 Subroutine: FEGS

Purpose: To calculate the performance index, the target errors, the targeting sensitivity matrix, and the performance gradient.

Method: FEGS provides the interface between the abstract control space targeting, and optimization search, and the actual low thrust trajectory generation. Trajectory parameters such as

1) Initial conditions
    - ecliptic state relative to primary body;
    - injection from parking orbit;
    - spacecraft mass;
2) Spacecraft engine characteristics;
3) Thrust controls;

are reset as specified by non-zero values of the H array (control perturbations). Subsequently, the trajectory propagator is called and trajectory information is collected.

Subroutine FEGS performs two major functions for TOPSEP depending upon the input value of IT. If IT equals 1, the target sensitivity matrix (S) and the performance gradient (G) are computed by finite differencing. A trajectory is generated for each perturbed control resulting in the computation of a column of the S matrix and an element of the G vector. The perturbations to the controls are input in PERT, a variable in the argument list. If IT is -1, a trial trajectory is generated. In this case all the specified trajectory parameters are reset before the trajectory propagator is called. After the trajectory is generated, the performance index (F) and the target errors (E) are evaluated. If IT is 0, a grid trajectory is generated. Basically the same logic flow is followed as for the trial trajectory generation. The primary differences are that only one element of PERT is non-zero and that no trajectory event times are stored in blank common.

Remarks: When the STM method of targeting is flagged (IASTM = 1) subroutine STMTAR constructs F, E, and S. Subroutine FEGS only generates the trial trajectories and the final reference trajectory.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
CHI	I/O	C	In plane ΔV direction angle at injection.
DELVO	I/O	C	Injection ΔV.
E	O	C	Target errors of the current trajectory.
ENGINE(1)	I/O	C	Power from solar array at 1 au.

Variable	Input/ Output	Argument/ Common	Definition
ENGINE(10)	I/O	C	Exhaust velocity.
F	I	C	Performance index.
FTR(1)	I	C	Performance index of the reference trajectory for the current iteration.
G	O	C	Performance gradient.
IT	I	A	1, generate perturbed trajectories and compute S and G 0, generate a grid trajectory and compute F and E -1, generate a trial trajectory and compute F and E.
ITERAT	I	C	Iteration counter (IT = 1 or -1); Control identifier for grid submode (IT = 0).
KMAX	I	C	Number of thrust controls (THRUST (I,J)) chosen to be elements of U.
LØCM	I	C	Blank common location of the current s/c mass.
LØCTS	I	C	Blank common location of event times for the reference and all trial trajectories in a single iteration.
NLP	I	C	Launch planet identifier (normally Earth).
NT	I	C	Number of targets.
NTR	I	C	Trial trajectory counter.

Variable	Input/ Output	Argument/ Common	Definition
NU	I	C	Number of controls.
PERT	I	A	Vector of control perturbations.
PSI	I/O	C	Out of plane ΔV direction angle at injection.
S	O	C	Target sensitivity matrix.
SCMASS	I/O	C	S/C mass corresponding to the trajectory start time (TSTART).
STATEO	I/O	C	S/C state corresponding to the trajectory start time (TSTART).
STATR	I/O	C	Array of initial states for the reference and all trial trajectories evaluated during the current iteration.
TARGET	I	C	Vector of desired target values.
TARNØM	O	C	Target values evaluated for the reference trajectory.
TARPAR	O	C	Target values of the most recently generated trajectory.
TARTR	I/O	C	Target values of the reference and all trial trajectories evaluated during a single iteration.
TM	I	C	Conversion constant: Number of seconds in a day.
TSTART	I/O	C	Trajectory start time.

Variable	Input/ Output	Argument/ Common	Definition
U	I	C	Selection of controls for the specified mode run.
VPARK	I/Ø	C	Parking orbit velocity at injection.
XMM	I/Ø	C	Mean motion of s/c in parking orbit.
PRO	I/Ø	C	Radial distance at injection.
PINC	I/Ø	C	Ecliptic inclination at injection.
PTO	I/Ø	C	Injection time relative to TLNCH.

Local Variables:

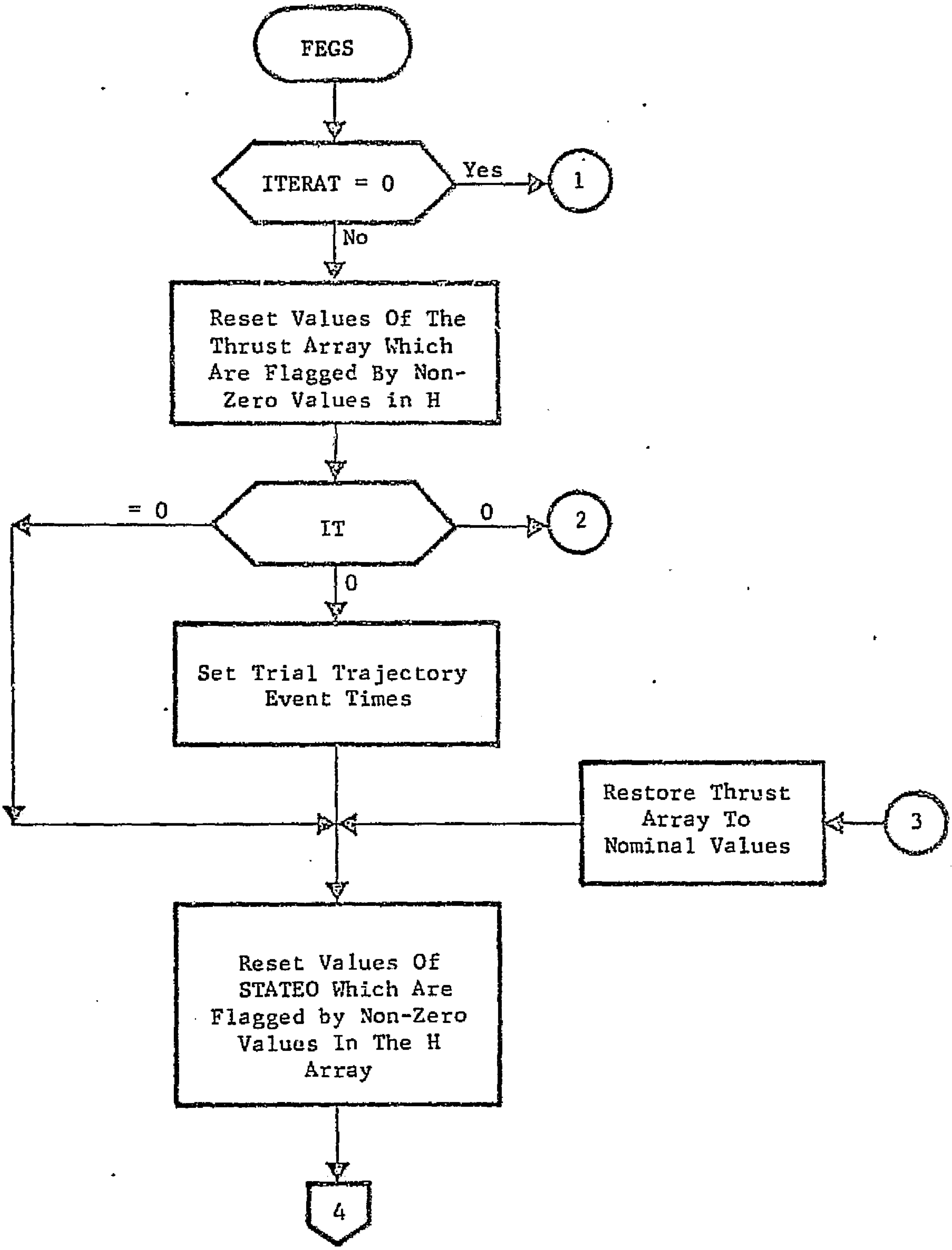
Variable	Definition
A	Semi-major axis of parking orbit.
CNTRØL	The nominal value of the control plus its perturbation.
ITRIAL	Trial step counter.
KALL	Statement number to which the logic flow returns after S and G are computed.
KØUNT	Control index.
ØMEGA	Longitude of the ascending node of the parking orbit.
SØMEGA	Argument of periapsis of the parking orbit.
XECC	Eccentricity of the parking orbit.
XMEAN	Mean anomaly of the s/c in parking orbit.

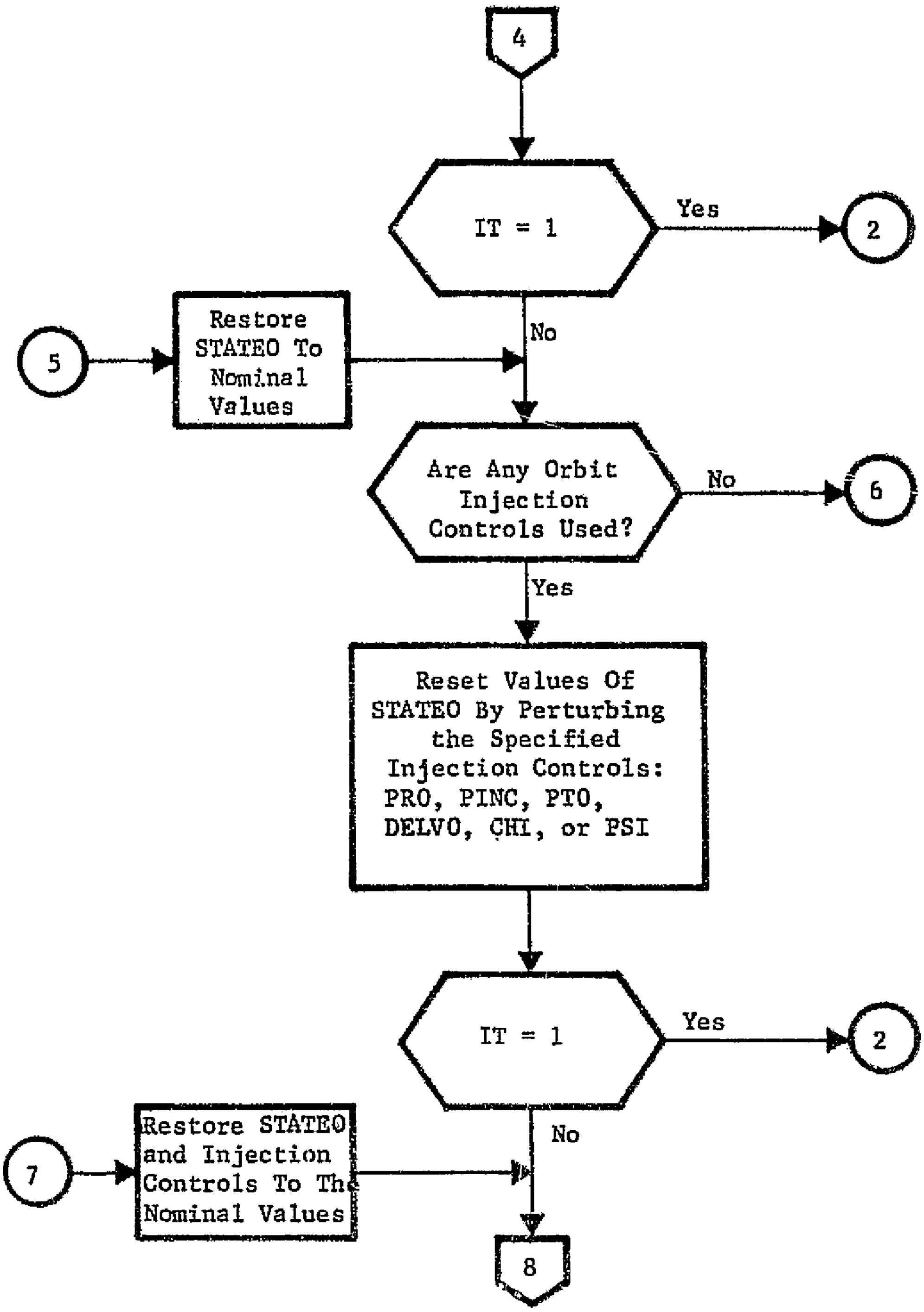
Subroutines Called: CARTES, CØNIC, CØPY, PRINT1, VECMAG, MATØUT, MODINJ

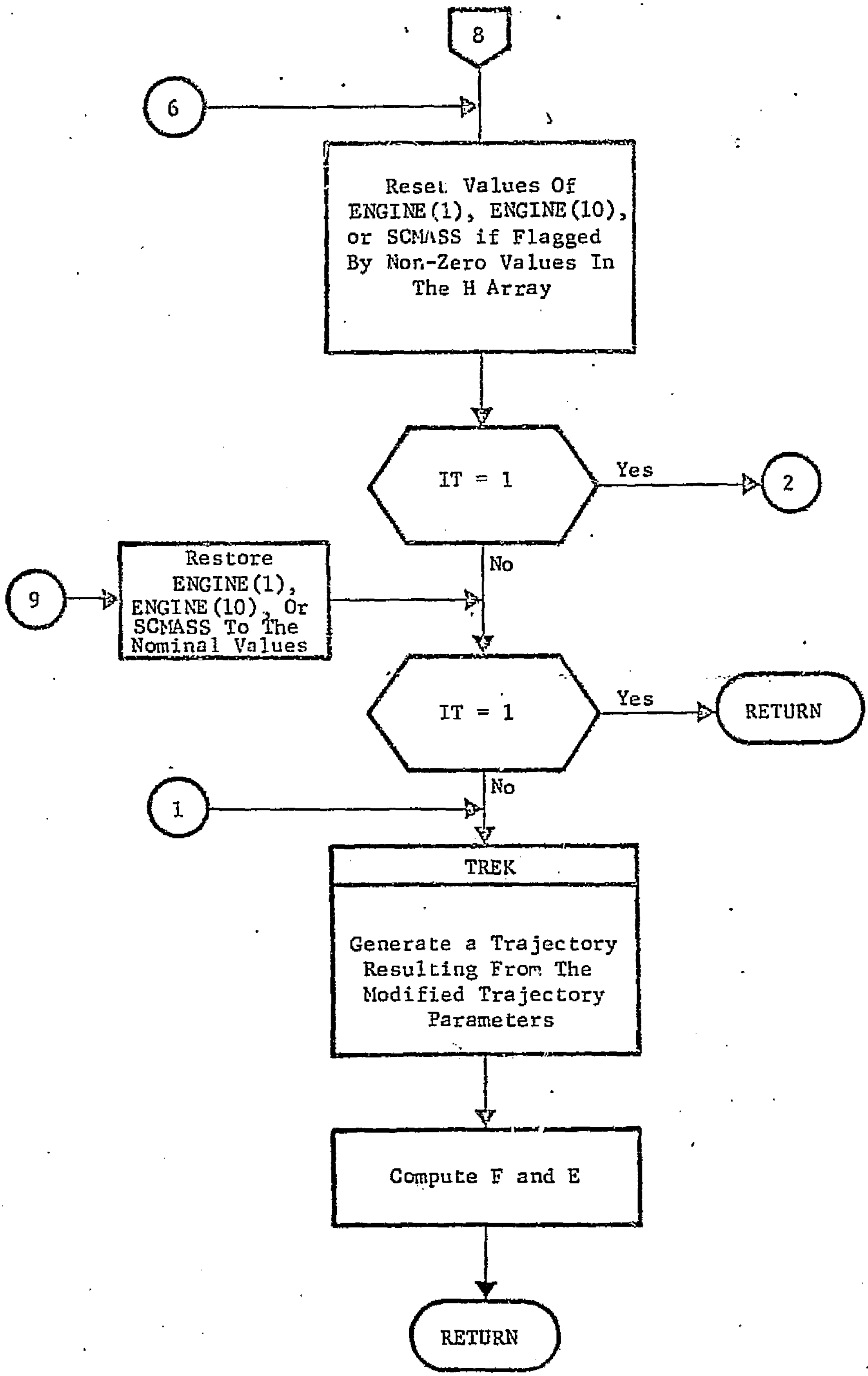
Calling Subroutines: GRID, PGM, TØPSEP

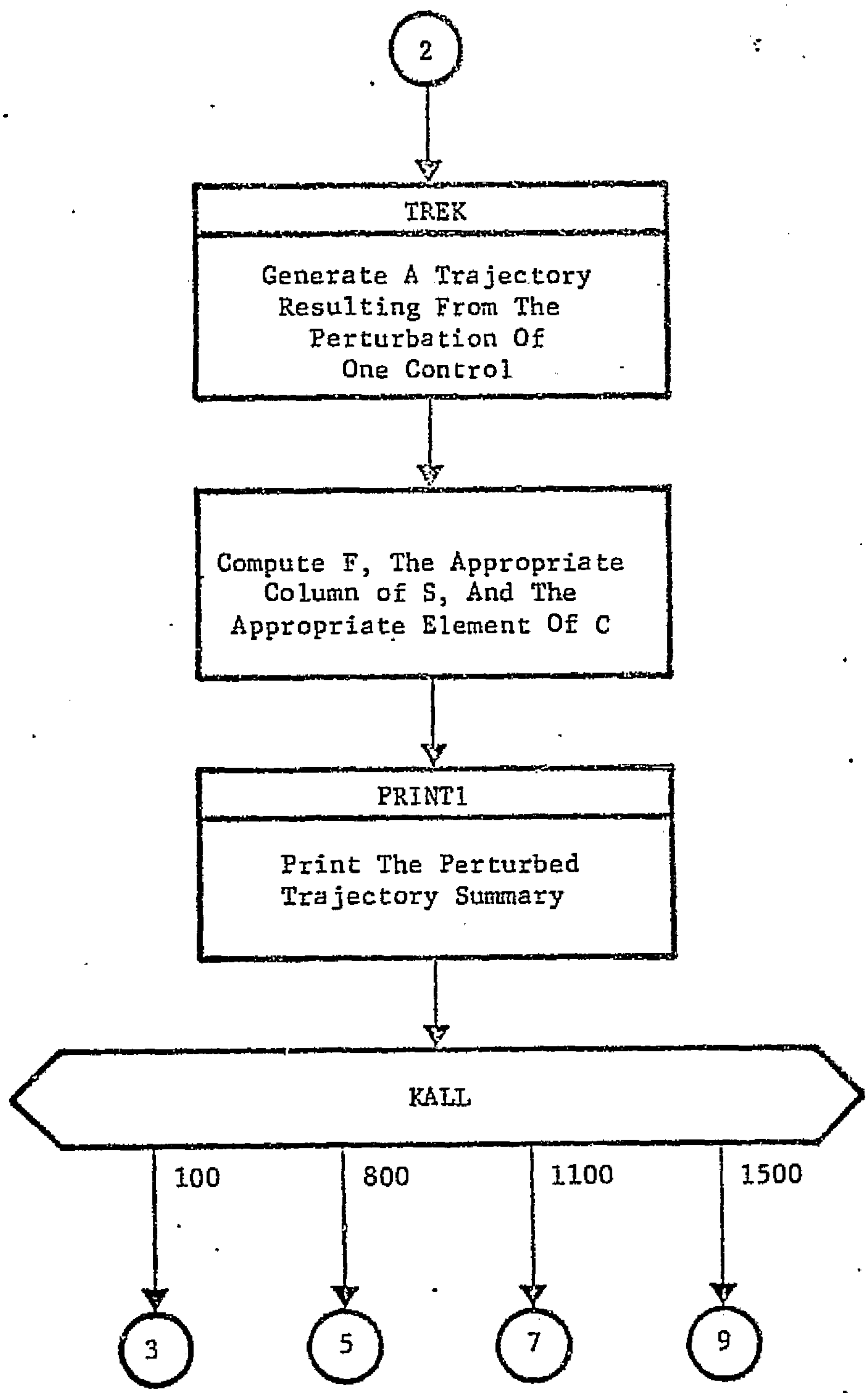
Common Blocks: (BLANK), CØNST, EDIT, EPHEM, TIME, TØP1, TØP2, TRAJ1, TRAJ2, WØRK

Logic Flow:


FEGS
ITERAT = 0
Yes
1
No
Reset Values Of The Thrust Array Which Are Flagged By Non-Zero Values in H
= 0
IT
0
2
0
Set Trial Trajectory Event Times
Restore Thrust Array To Nominal Values
3
Reset Values Of STATEO Which Are Flagged by Non-Zero Values In The H Array
4


4
IT = 1
Yes
2
No
5
Restore STATEO To Nominal Values
Are Any Orbit Injection Controls Used?
No
6
Yes
Reset Values Of STATEO By Perturbing the Specified Injection Controls: PRO, PINC, PTO, DELVO, CHI, or PSI
IT = 1
Yes
2
No
7
Restore STATEO and Injection Controls To The Nominal Values
8


8
6
Reset Values Of ENGINE(1), ENGINE(10), or SCMASS if Flagged By Non-Zero Values In The H Array
IT = 1
Yes
2
No
9
Restore ENGINE(1), ENGINE(10), Or SCMASS To The Nominal Values
IT = 1
Yes
RETURN
No
1
TREK
Generate a Trajectory Resulting From The Modified Trajectory Parameters
Compute F and E
RETURN


2
TREK
Generate A Trajectory Resulting From The Perturbation Of One Control
Compute F, The Appropriate Column of S, And The Appropriate Element Of C
PRINT1
Print The Perturbed Trajectory Summary
KALL
100
800
1100
1500
3
5
7
9

3.2.5 Subroutine: FGAMA (IS)

Purpose: To evaluate the net cost index and target error index of a trial trajectory.

Method: Subroutine FGAMA scales the control correction $\Delta \underline{u}$ by GAMMA(NTR), which is computed in GENMIN, and calls FEGS to generate a trial trajectory. Preceding the call to FEGS for the second trial trajectory generation, a computation is made to estimate the scale factor which will reduce the value of the final spacecraft mass to some specified limit (FTR(1) - DF). This scale factor becomes the maximum allowable scale for future trial steps, unless the scale is further restricted by explicit control bounds. However, no additional constraint is placed on the scale factor if the final spacecraft mass is increased by taking larger trial steps in the $\Delta \underline{u}$ direction. The scale factor is not restricted due to the performance constraint prior to the second trial step for lack of information to make an accurate estimate.

Remarks: The cost index F is actually the negative of the final spacecraft mass. If the cost index is decreasing (becoming more negative) in the $\Delta \underline{u}$ direction the estimation loop is bypassed.

If the loop must be entered because the cost is increasing, a modification must be made to the cost index values (FTR) so that the routines MINMUM and THPM may be used. To find the minimum value of the final spacecraft mass the negative of the cost index is minimized in the $\Delta \underline{u}$ direction.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
DFMAX	I	C	Maximum percentage decrease allowed in the s/c final mass for iteration.
E	O	C	Target errors of the current trajectory.
ETØL	I	C	Target tolerances.
ETR	I/O	C	Array of target errors of the reference and all trial trajectories evaluated during a single iteration.
F	O	C	Cost index of the current trajectory.
FTR	I/O	C	Vector of cost indices of the reference and all trial trajectories evaluated during a single iteration.
G	O	C	Performance gradient.
GAMMA	I	C	Vector of trial trajectory control change scale factors.
GTRIAL(2)	I/O	C	Maximum allowable value for GAMMA.

Variable	Input/ Output	Argument/ Common	Definition
IS	I	A	Trial trajectory number.
LØCDU	I	C	Blank common location of the control correction vector $\Delta \underline{u}$.
LØCSDU	I	C	Blank common location of the trial step (GAMMA(NTR)* $\Delta \underline{U}$); used as such only when generating trial trajectories.
LØCSI	I	C	Blank common location of the pseudo inverse of the weighted sensitivity matrix.
NT	I	C	Number of targets.
NTR	O	C	Trial trajectory counter (NTR = 1 for the iteration reference trajectory).
NU	I	C	Number of controls.
ØSCALE	I	C	Scale on the net cost index P1 when simultaneously targeting and optimizing.
P1	O	C	Vector of net cost values for the reference and all trial trajectories evaluated during a single iteration.
P1P2	O	C	Vector of combined target error indices and net cost values.
P2	O	C	Vector of target error indices for the reference and all trial trajectories evaluated during a single iteration.
TARPAR	O	C	Target values of the most recently generated trajectory.

Variable	Input/ Output	Argument/ Common	Definition
WE	I	C	Vector of target weights.

Local Variables:

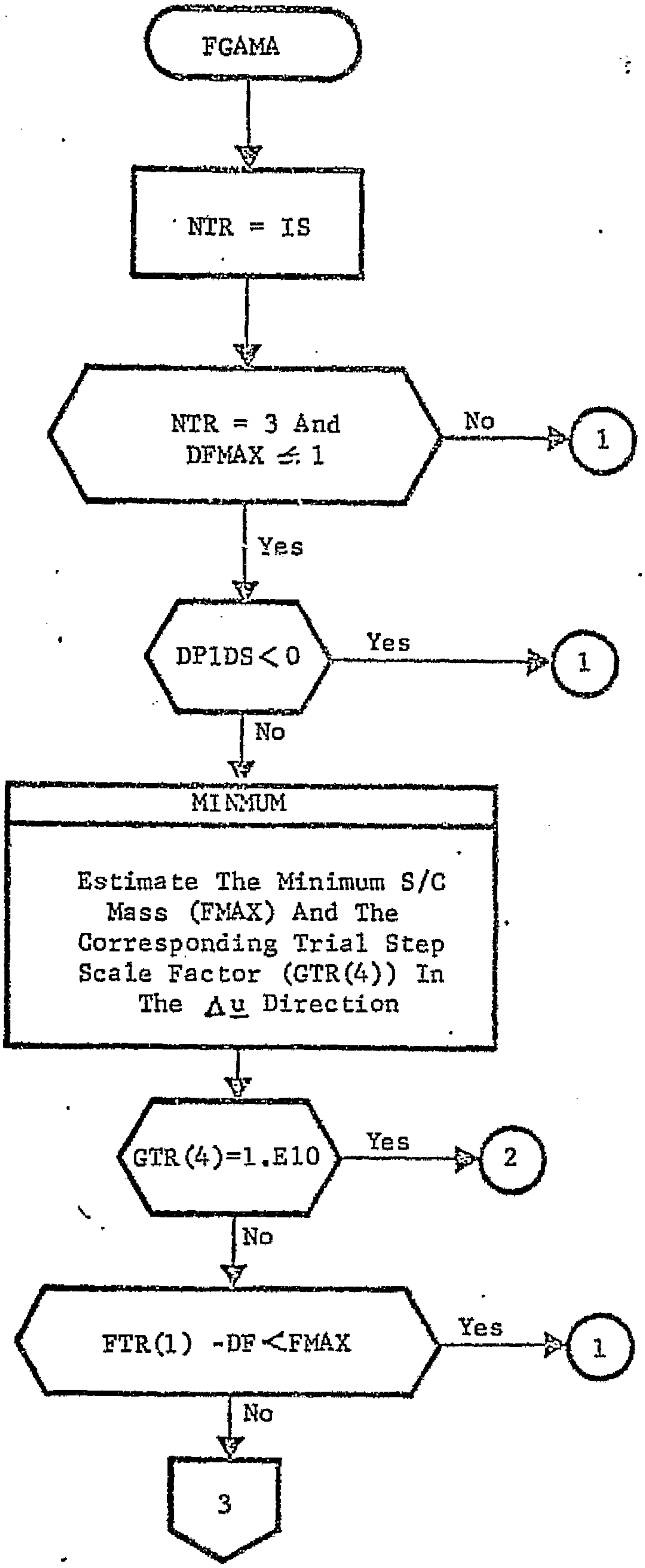
Variable	Definition
DF	Maximum decrease allowed in the final s/c mass.
DP1DS	First derivative of P1 evaluated at GAMMA(1) = 0.
EPRIME (=WØRK(1))	Vector of target errors divided by tolerances.
FMAX	Estimated maximum cost evaluated in the $\Delta \underline{u}$ direction.
FTEST (=WØRK(55))	Vector of cost indices corresponding to the scale factors GTR(I), I = 1, 3 where GTR(1) < GTR(2) < GTR(3).
GDU (=WØRK(13))	Linearized approximation to change in cost function required to perform a minimum - norm correction back to the targeted manifold.
GTR(1) (=WØRK(50))	GAMMA(1).
GTR(2) (=WØRK(51))	MIN { GAMMA(2), GTR(4) }
GTR(3) (=WØRK(52))	MAX { GAMMA(2), GTR(4) }
GTR(4) (=WØRK(53))	Scale factor corresponding to FMAX.
GTS (=WØRK(7))	Intermediate storage in GDU computation.
IERR	Flag set to 1 to direct MINMUM and THPM to compute GTR(4) given F(GTR(4)) using the prescribed polynominal expansion.

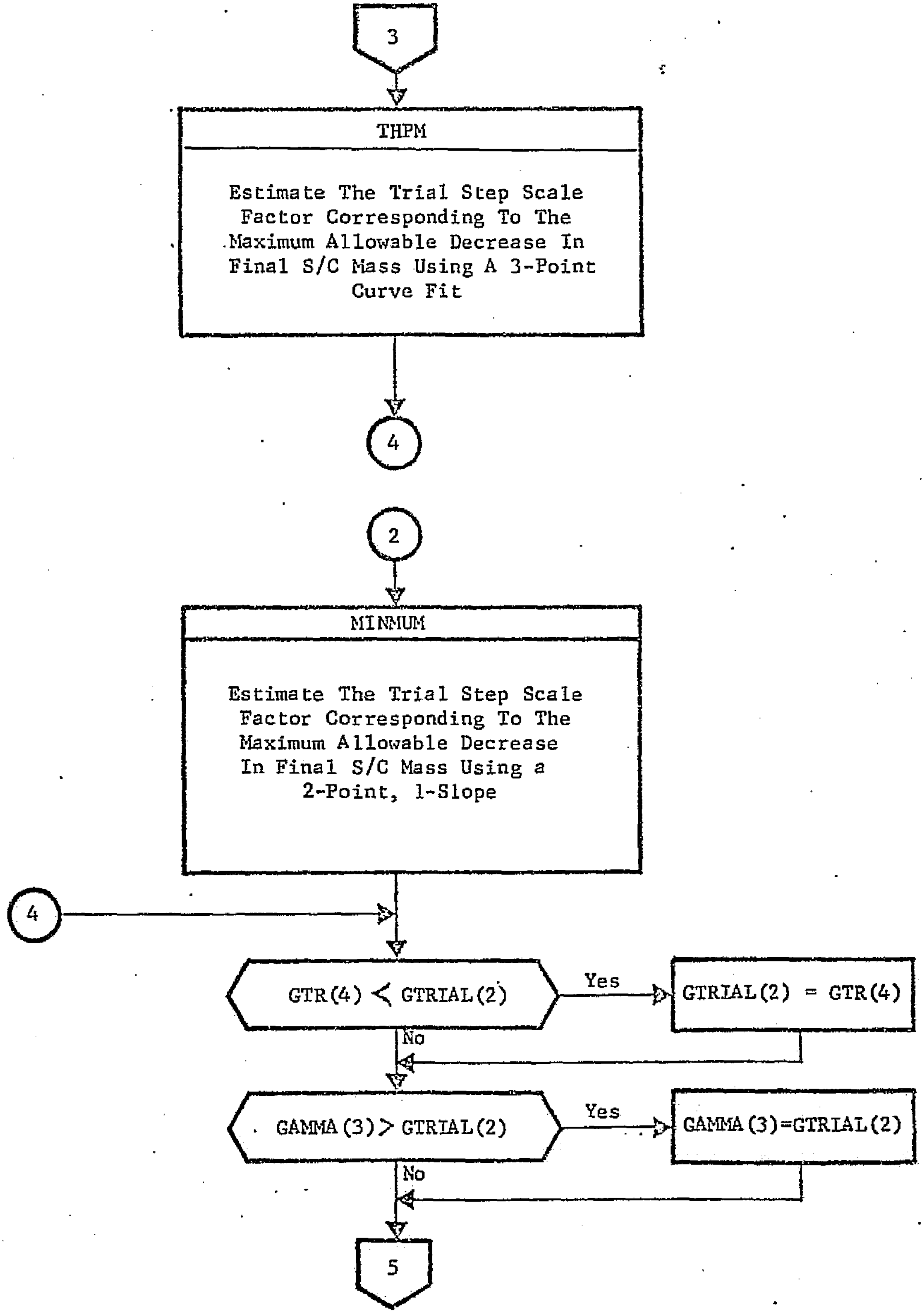
<u>Subroutines Called</u>: CØPY, FEGS, MATØUT, MINMUM, MMAB, MMATB, MMATBA, NEGMAT, SCALE, THPM, ZERØM

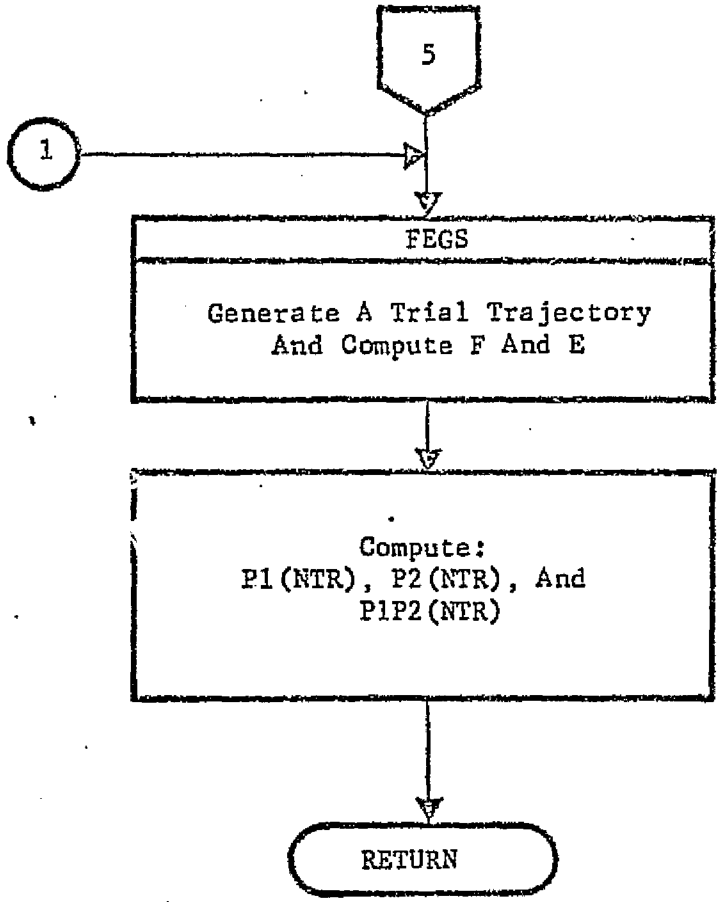
<u>Calling Subroutines</u>: GENMIN

<u>Common Blocks</u>: (BLANK), EDIT, TØP1, TØP2, WØRK

Logic Flow:


FGAMA
NTR = IS
NTR = 3 And DFMAX ≤ 1
No
1
Yes
DP1DS < 0
Yes
1
No
MINMUM
Estimate The Minimum S/C Mass (FMAX) And The Corresponding Trial Step Scale Factor (GTR(4)) In The Δu Direction
GTR(4)=1.E10
Yes
2
No
FTR(1) -DF < FMAX
Yes
1
No
3


3
THPM
Estimate The Trial Step Scale Factor Corresponding To The Maximum Allowable Decrease In Final S/C Mass Using A 3-Point Curve Fit
4
2
MINMUM
Estimate The Trial Step Scale Factor Corresponding To The Maximum Allowable Decrease In Final S/C Mass Using a 2-Point, 1-Slope
4
GTR(4) < GTRIAL(2)
Yes
GTRIAL(2) = GTR(4)
No
GAMMA(3) > GTRIAL(2)
Yes
GAMMA(3)=GTRIAL(2)
No
5


5
1
FEGS
Generate A Trial Trajectory
And Compute F And E
Compute:
P1(NTR), P2(NTR), And
P1P2(NTR)
RETURN

3.2.6 Subroutine: GENMIN (X, Y, DYDX1, GTRIAL, YES, MIN)

Purpose: To choose the best control change scale factor based on a one-dimensional search in the new control vector direction.

Remarks: The best scale factor will be defined as that which provides for the minimum value of the net cost-function as described in subroutine SIZE. The one dimensional search will consist of a series of second and third order polynomial curve fitting techniques.

Input/Output:

Variable	Input/ Output	Argument(A)/ Common(C)	Definition
DYDX1	I	A	Value of the first derivative of the net cost function evaluated at X(1)=0
GTRIAL(1)	I	A	If X(I+1) < GTRIAL(1)*X(I), then X(I+1) is set equal to GTRIAL(1) *X(I)
GTRIAL(2)	I	A	Maximum allowable scale factor value
GTRIAL(3)	I	A	The percentage of X(I+1) to X(I) above which the search will be terminated.
GTRIAL(4)	I	A	The percentage of YES(I) to Y(I+2) below which the search is terminated
GTRIAL(5)	I	A	Flag designating the extent of curve fitting in the new control direction (i.e., GTRIAL(5)=4 signifies all four techniques may be used)
MIN	Ø	A	Pointer designating the minimizing scale factor
X(1)	I	A	X(1)=0, value of scale factor associated with current net cost function value

Input/Output: - Continued

Variable	Input/ Output	Argument(A)/ Common(C)	Definition
X(2)	I	A	Value of scale factor for first trial net cost-function evaluation
X(3)	Ø	A	Scale factor returned from "two point, one slope" curve fitting routine
X(4)	Ø	A	Scale factor returned from "three point, one slope" curve fitting routine
X(5)	Ø	A	Scale factor returned from "three point" curve fitting routine
X(6)	Ø	A	Scale factor returned from "four point" curve fitting routine
Y(1)	I	A	Value of current net cost-function
Y(2) → Y(6)	Ø	A	Trial net cost-function values associated with X(2)→X(6)
YES	Ø	A	Vector of estimates of net cost-function values returned from the curve fitting routines

Local Variables:

Variable	Definition
MAX	The number of trial net cost-function values which must be tested for the local minima
MINSV	The number of a trial net cost-function value which is a local minimum but not necessarily the global minimum

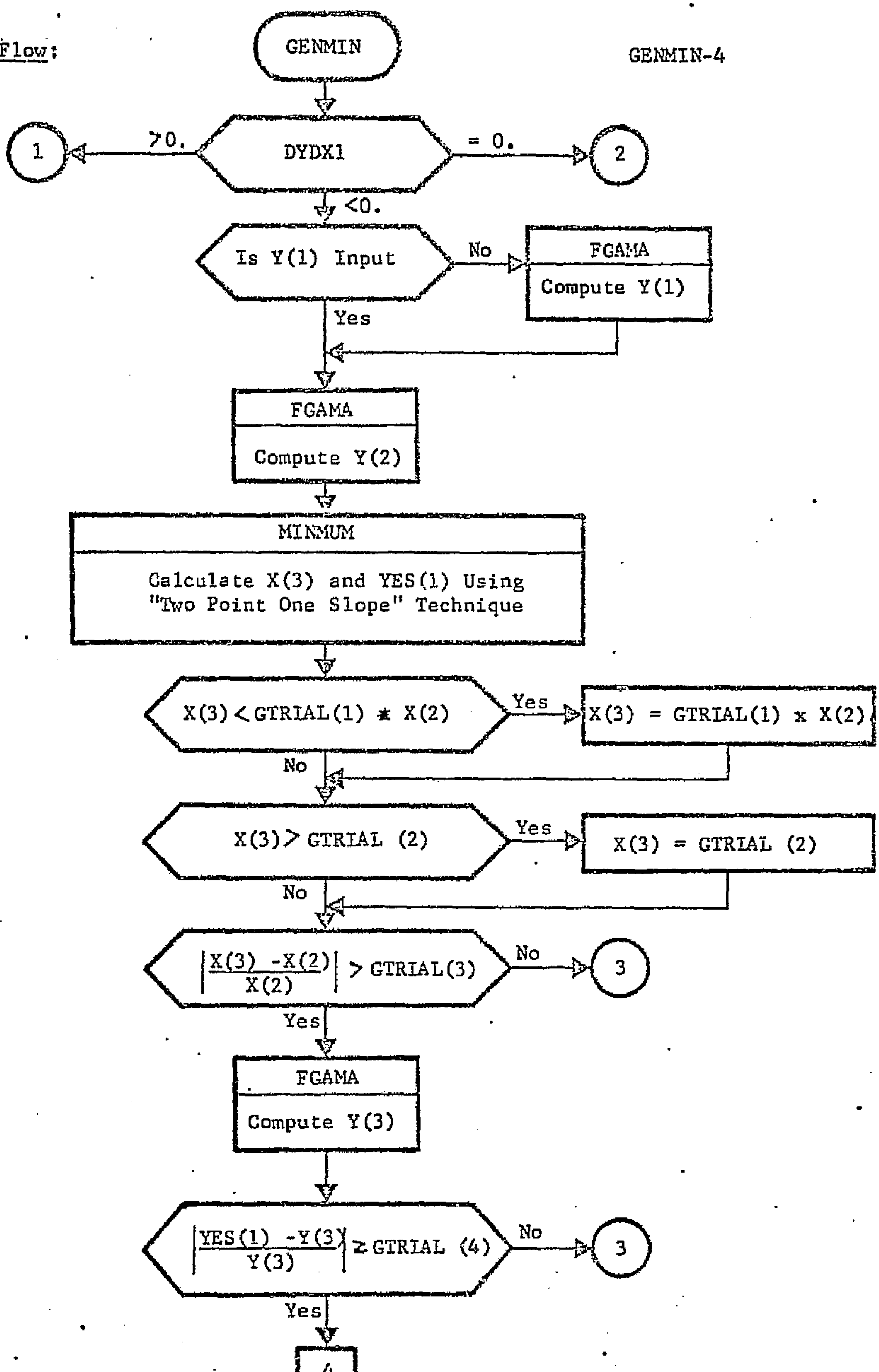
<u>Subroutines Called</u>: BUCKET, FGAMA, MINMUM

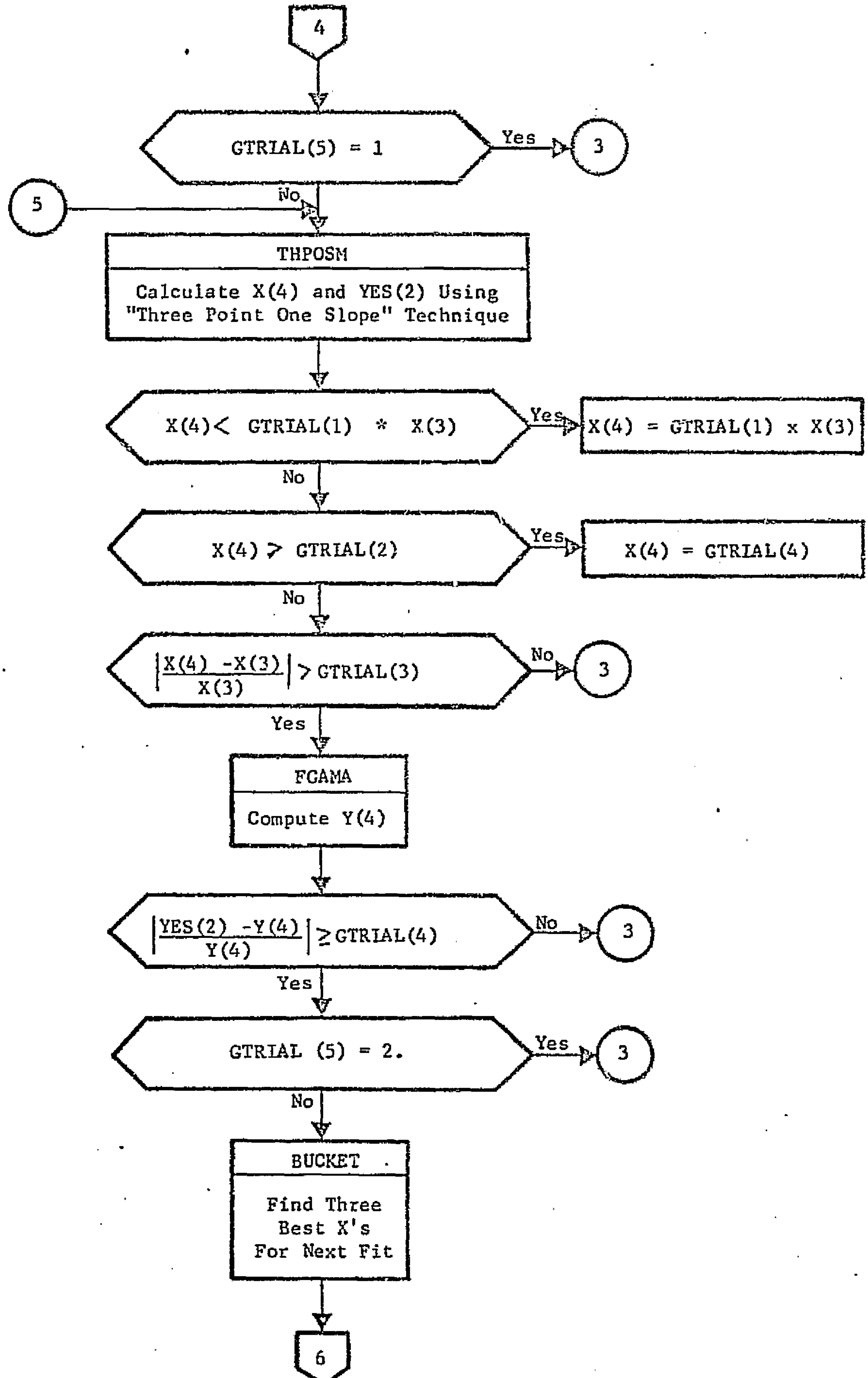
<u>Calling Subroutines</u>: SIZE

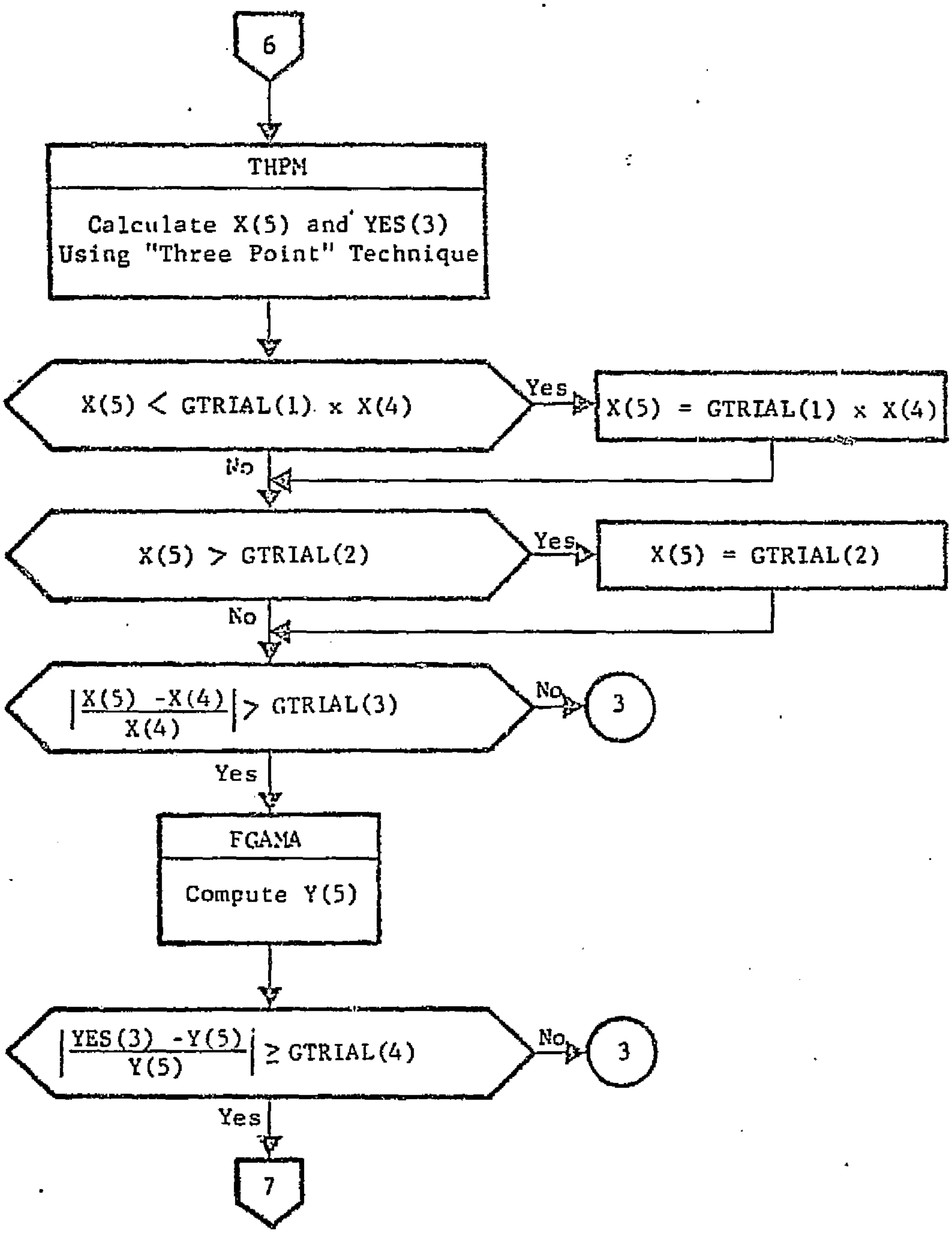
<u>Common Blocks</u>: None

Logic Flow:

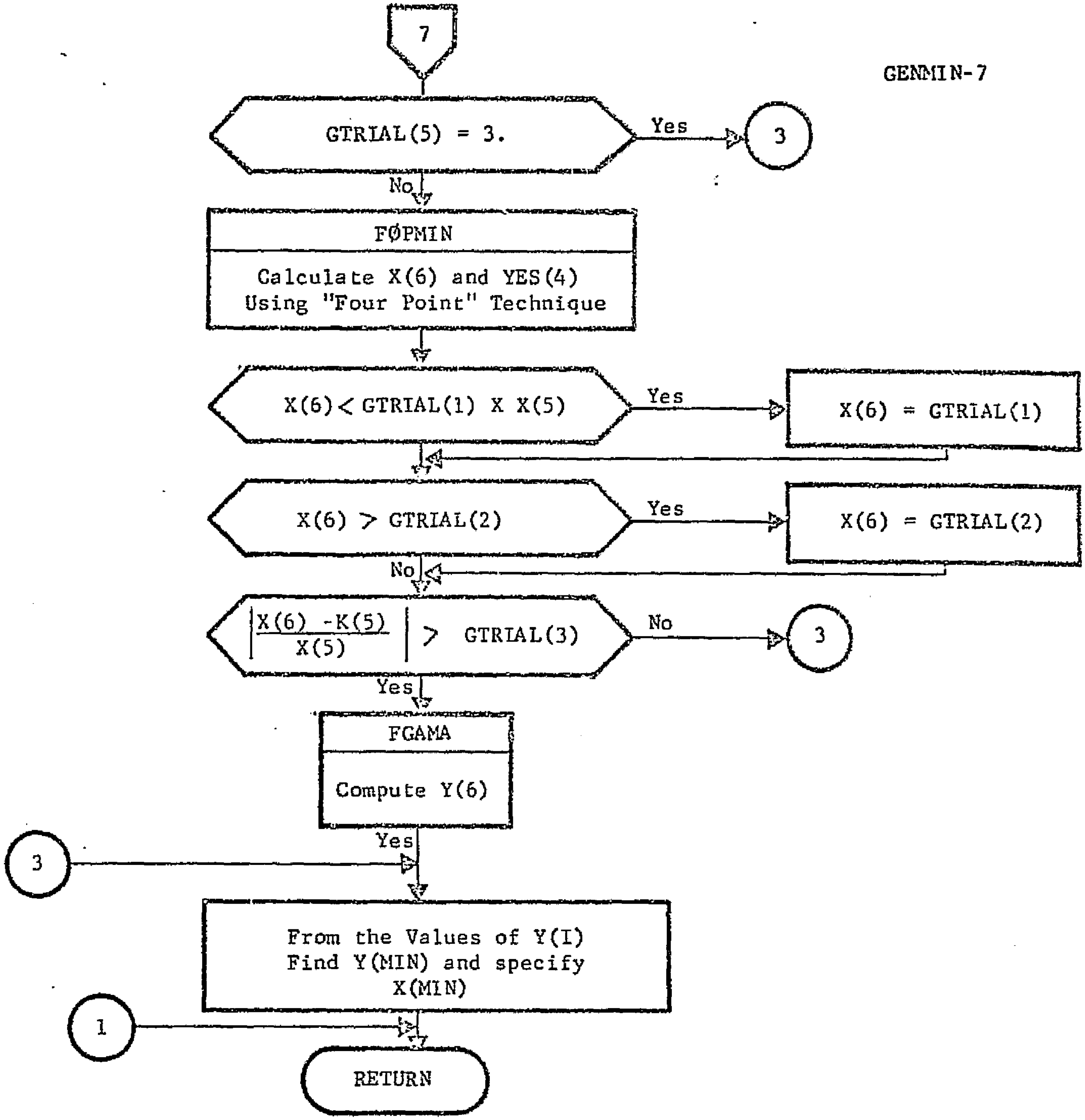
GENMIN-4



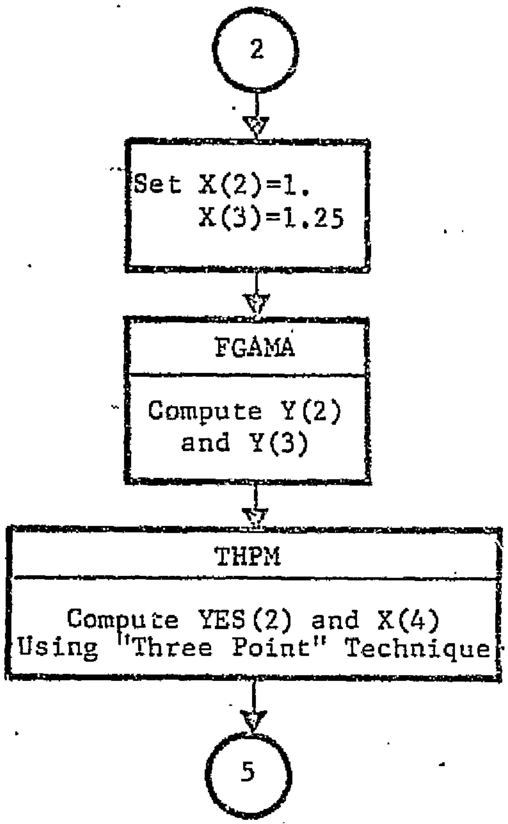

4
GTRIAL(5) = 1
Yes
3
5
No
THPOSM
Calculate X(4) and YES(2) Using "Three Point One Slope" Technique
X(4)< GTRIAL(1) * X(3)
Yes
X(4) = GTRIAL(1) x X(3)
No
X(4) > GTRIAL(2)
Yes
X(4) = GTRIAL(4)
No
|X(4) -X(3) / X(3)| > GTRIAL(3)
No
3
Yes
FGAMA
Compute Y(4)
|YES(2) -Y(4) / Y(4)| ≥ GTRIAL(4)
No
3
Yes
GTRIAL (5) = 2.
Yes
3
No
BUCKET
Find Three Best X's For Next Fit
6


6
THPM
Calculate X(5) and YES(3) Using "Three Point" Technique
X(5) < GTRIAL(1) x X(4)
Yes
X(5) = GTRIAL(1) x X(4)
No
X(5) > GTRIAL(2)
Yes
X(5) = GTRIAL(2)
No
|(X(5) -X(4))/X(4)| > GTRIAL(3)
No
3
Yes
FGAMA
Compute Y(5)
|(YES(3) -Y(5))/Y(5)| ≥ GTRIAL(4)
No
3
Yes
7

GENMIN-7



7
GTRIAL(5) = 3.
Yes
3
No
FØPMIN
Calculate X(6) and YES(4)
Using "Four Point" Technique
X(6) < GTRIAL(1) X X(5)
Yes
X(6) = GTRIAL(1)
X(6) > GTRIAL(2)
Yes
X(6) = GTRIAL(2)
No
|X(6) -K(5) / X(5)| > GTRIAL(3)
No
3
Yes
FGAMA
Compute Y(6)
Yes
3
From the Values of Y(I)
Find Y(MIN) and specify
X(MIN)
1
RETURN


2
Set X(2)=1.
X(3)=1.25
FGAMA
Compute Y(2)
and Y(3)
THPM
Compute YES(2) and X(4)
Using "Three Point" Technique
5

3.2.7A Subroutine: GRID

Purpose: To generate a family of trajectories in order to obtain performance and error index information.

Method: Consider an NU-dimensional control space and a nominal control vector u. A grid of trajectory target error indices and performance indices is generated based upon two steps from the nominal control vector in each control direction. The first step in the i^{th} control direction is specified by the i^{th} element of PRTURB. The second step for the same control is specified by $HMULT_i$ * $PRTURB_i$.

Remarks: The user can take advantage of the cycling capability of the TOPSEP mode to specify more than two steps in each of the control directions by stacking cases.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
CNVRTT	I	C	Conversion constants from internal target units to output target units.
E	I	C	Target errors of current trajectory.
ETR(1, 1)	O	C	Target error index of nominal trajectory.

Variable	Input/ Output	Argument/ Common	Definition
F	I	C	Performance index of current trajectory.
FTR(1)	O	C	Performance index of nominal trajectory.
HMULT	I	C	Vector containing the scale on the elements of PRTURB for the second step in each control direction.
ITERAT	O	C	Index specifying which control element is being changed.
KØNVRJ	O	C	Index specifying the step number in the control direction under consideration.
LABELT	I	C	Hollerith labels for specified targets.
LØCDU1	I	C	Location in blank common of the first control steps.
LØCDU2	I	C	Location in blank common of the second control steps.
LØCEM1	I	C	Location in blank common of the target error indices associated with the first control steps.
LØCEM2	I	C	Location in blank common of the target error indices associated with the second control steps.
LØCEN	I	C	Location in blank common of the target errors of the nominal trajectory.
LØCE1	I	C	Location in blank common of the target errors associated with the first control steps.

Variable	Input/ Output	Argument/ Common	Definition
LØCE2	I	C	Location in blank common of the target errors associated with the second control steps.
LØCF1	I	C	Location in blank common of the performance indices associated with the first control steps.
LØCF2	I	C	Location in blank common of the performance indices associated with the second control steps.
NT	I	C	Number of targets.
NTR	I	C	Flag used to set the branch of logic followed in FEGS (always set to 1).
NU	I	C	Number of controls.
PRTURB	I	C	Perturbations to the controls for the first step in each control direction.
STØRE	I	C	Blank common variable for storage.
WE	I	C	Vector used to compute target error index, containing $\frac{1}{\text{TARTØL(I)}^2}$
WØRK	I	C	Working storage.

Local Variables:

Variable	Definition
PERT (= UWATE)	Vector used to transfer the control steps to FEGS where F and E are computed.

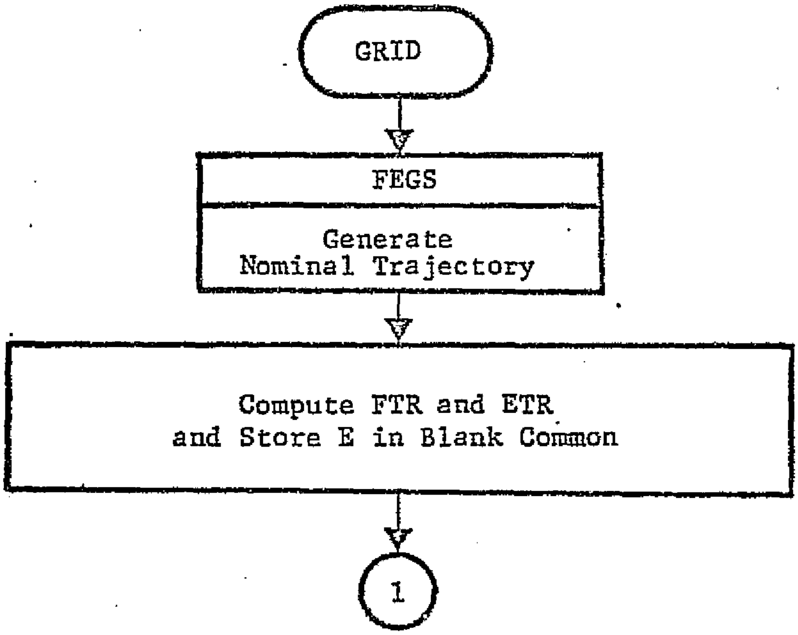
Variable	Definition
WETØL (= S)	Array whose off-diagonal elements are zero and whose diagonal elements are WE(I)

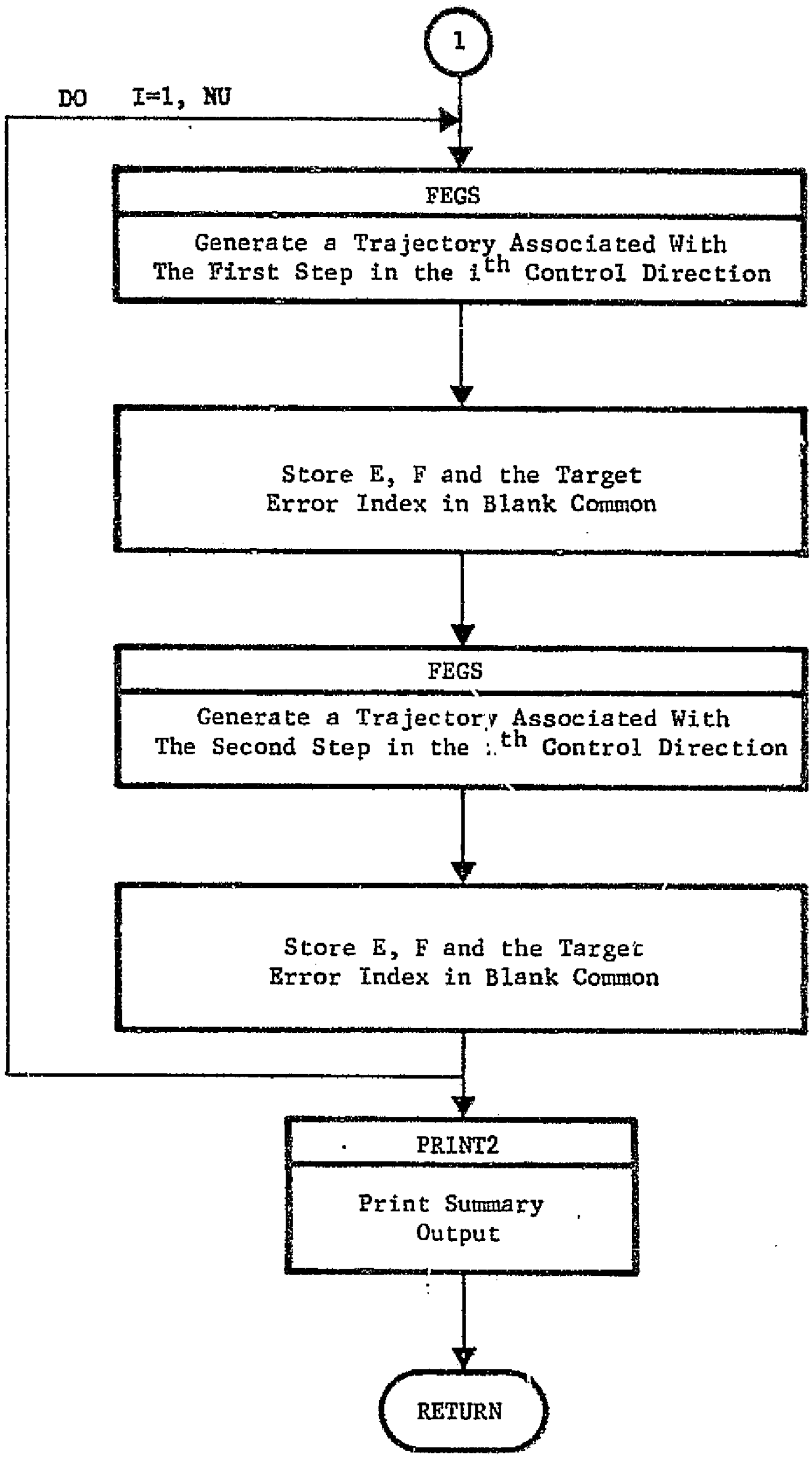
Subroutines Called: CØPY, FEGS, MMATBA, PRINT2, ZERØM

Calling Subroutines: TØPSEP

Common Blocks: (BLANK), EDIT, GRID, PRINTH, TØP1, TØP2, WØRK

Logic Flow:




1
DO I=1, NU
FEGS
Generate a Trajectory Associated With The First Step in the ith Control Direction
Store E, F and the Target Error Index in Blank Common
FEGS
Generate a Trajectory Associated With The Second Step in the ith Control Direction
Store E, F and the Target Error Index in Blank Common
PRINT2
Print Summary Output
RETURN

3.2.7B Subroutine: INJECT

Entry Points: MØDINJ
TUGINJ

Purpose: To generate tug injection data

Method: The analytic discussion of the injection process may be found in Reference 1, Section 9.5, page 124.

Remarks: Subroutine INJECT consists of three related computational blocks. Each block corresponds to an entry point.

- INJECT, computation of outer parking orbit and injection parameters: PRO, PINC, PTO, DELVO, CHI, and PSI.
- MØDINJ, computation of the initial state based upon perturbed injection parameters.
- TUGINJ, computation of inner parking orbit and fuel requirements for the parking orbit transfer.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
AZMAX	I	C	Maximum launch azimuth constraint.
AZMIN	I	C	Minimum launch azimuth constraint.
CHI	I/O	C	In-plane ΔV direction angle at injection.
DELVO	I/O	C	ΔV at injection.
ECEQ	I	C	Transformation matrix from Earth equatorial to ecliptic.
H	I	C	Array of control perturbations.

Variable	Input/ Output	Argument/ Common	Definition
INJLØC	I	C	Location of injection parameters in control vector.
NLP	I	C	Launch planet designation.
PINC	I/O	C	Ecliptic inclination of outer parking orbit.
PMASS	I	C	Vector of planetary masses.
PRO	I/O	C	Geocentric radial distance to S/C at injection.
PSI	I/O	C	Out-of-plane ΔV direction angle at injection.
PTO	I/O	C	Injection time relative to launch epoch.
RAD	I	C	Angle conversion constant (radians to degrees).
RP1	I	C	Inner parking orbit radius.
SCMASS	I	C	Initial S/C mass.
STATEO	I/O	C	Initial S/C state.
TGFUEL	I	C	Fuel capacity of tug vehicle.
TUG	I	C	Logical flag specifying injection computations if TRUE.
TUGISP	I	C	Specific impulse of tug vehicle.
TUGWT	I	C	Dry weight of tug vehicle.
U	I	C	Control vector.
VPARK	I/O	C	Parking orbit velocity at injection.
XMM	I/O	C	S/C mean motion in outer parking orbit.

Local Variables:

Variable	Definition
ANGLE (= WØRK(30))	Plane change required during parking orbit transfer.
DELVA (= WØRK(32))	First impulsive ΔV.
DELVB (= WØRK(33))	Second impulsive ΔV.
DELV1 (= WORK(41))	Single maneuver injection ΔV.
EC (= WØRK(40))	Eccentricity of hyperbolic escape orbit for single maneuver trajectory.
EQIMAX (= WORK(28))	Maximum equatorial inclination constraint.
EQIMIN (= WORK(29))	Minimum equatorial inclination constraint.
EQI1 (= WØRK(31))	Equatorial inclination of inner parking orbit.
EQI2 (= WØRK(27))	Equatorial inclination of outer parking orbit.
GRAV	Gravitational constant.
PHILAT	Latitude of launch site.
STATEQ (= WORK(21))	Initial state in equatorial coordinates.
VINF (= WORK(39))	Hyperbolic excess velocity
WFUELA (= WORK(35))	Fuel required for first tug maneuver.
WFUELB (= WORK(36))	Fuel required for second tug maneuver.
WFUELT (= WØRK(38))	Total fuel requirement.
WFUELO (= WORK(37))	Fuel required for third tug maneuver (injection).
WFUEL1 (= WORK(42))	Fuel required for single tug maneuver (injection) from inner parking orbit.
WTØT (= WORK(34))	Total tug weight plus payload prior to any maneuvers.
XECC	Eccentricity of outer parking orbit.

Subroutines Called: ADD, MMATB, SCALE, UDOTV, UNITV, UXV, VECMAG

Calling Subroutines: PGM, FEGS, TREK, STMTAR

Common Blocks: CØNST, EPHEM, TØP1, TØP2, TRAJ1, TRAJ2, TUG, WØRK

Logic Flow: See listing

3.2.8 Subroutine: MINMUM (X, Y, DYDX1, XMIN, YMIN, IERR)

Entry Points:

THPM

THPØSM

FØPMIN

Purpose: To estimate a local minimum of the cost function Y(X) and the minimizing independent variable X* by fitting selected sample points with a quadratic or cubic polynomial.

Input/Output:

Variable	I/O	Argument (A)/ Common (c)	Definition
DYDX1	I	A	Value of the first derivative of Y with respect to X evaluated at X(1) = 0.
IERR	0	A	Flag whose non-zero value indicates that two of the given X values are identical.
X	I	A	Vector of independent variable sample values
XMIN	0	A	Minimizing independent variable X*
Y	I	A	Vector of cost function sample values
YMIN	0	A	Local minimum of the cost function, y(X*)

Local Variables:

Variable	Definition
A	Cubic polynomial coefficients

Subroutines Called: None

Calling Subroutines: GENMIN, FGAMA

Common Blocks: None

Method: The function Y(X) is approximated by either a second or third order polynomial in order to compute analytically the minimizing parameter X*. The polynomial approximation is of the form

$$Y(X) \cong P(X) = \sum_{i=0}^{n} a_i X^i$$

where n = 2 or n = 3. The following four cases describe the method of approximation and the resulting minimization process

Case 1: Y is fitted with a quadratic polynomial based on

1) $Y(0)$

2) $\frac{dY}{dX}\Big|_{X=0}$

3) $Y(X_o)$ where $X_o > 0$ is an initial estimate of X*

The quadratic polynomial coefficients are calculated from the formulae

$$a_o = Y(0)$$

$$a_1 = \frac{dY}{dX}\Big|_{X=0}$$

$$a_2 = \frac{Y(X_o) - a_o}{X_o^2} + \frac{a_1}{X_o}$$

The independent variable value minimizing the quadratic is

$$X^* = \frac{-a_1}{2a_2}$$

Case 2

Y is fitted with a cubic polynomial based on:

1) $Y(0)$

2) $\left.\frac{dY}{dX}\right|_{X=0}$

3) $Y(X_o)$ where $X_o > 0$ is a sample value

4) $Y(X_1)$ where $X_1 > 0$ is a sample value

The cubic polynomial coefficients are calculated from the following formula:

$$\lambda = \max\{X_o, X_1\}$$

$$\alpha = \min\{X_o, X_1\} / \lambda$$

$$a_o = Y(0)$$

$$a_1 = \left.\frac{dY}{dX}\right|_{X=0}$$

$$a_2 = \left[\frac{Y(\lambda\alpha) - \alpha^3 Y(\lambda)}{1-\alpha} - \lambda\alpha(1+\alpha)\, a_1 - (1+\alpha+\alpha^2)\, a_o\right](\lambda^2\alpha^2)^{-1}$$

$$a_3 = \left[\lambda\alpha a_1 + a_o(1+\alpha) + \frac{\alpha^2 Y(\lambda) - Y(\alpha\lambda)}{1-\alpha}\right](\lambda^3\alpha^2)^{-1}$$

The independent variable value, X* minimizing P is

$$X^* = \left[-a_2 + \sqrt{a_2^{\,2} - 3a_3\, a_1}\right](3a_3)^{-1}$$

<u>Case 3</u>

A quadratic polynomial is fitted to $Y(X_o)$, $Y(X_1)$, $Y(X_2)$ where X_o, X_1, X_2 are greater than or equal to zero and represent sample values of X (not necessarily the same values as in prior cases). It is assumed that:

1) $X_o \leq X_1 < X_2$

2) $Y(X_o) > Y(X_1) < Y(X_2)$

The formulae for the quadratic coefficients are as follows:

$$b_{ij} = X_i \, X_j$$

$$c_{ij} = X_i + X_j$$

$$d_{ij} = X_i - X_j$$

$$a_o = \frac{b_{12}}{d_{01}d_{02}} Y(X_o) + \frac{b_{02}}{d_{10}d_{12}} Y(X_1) + \frac{b_{01}}{d_{20}d_{21}} + Y(X_2)$$

$$a_1 = -\frac{c_{12}}{d_{01}d_{02}} Y(X_o) - \frac{c_{02}}{d_{10}d_{12}} Y(X_1) - \frac{c_{01}}{d_{20}d_{21}} Y(X_2)$$

$$a_2 = \frac{Y(X_o)}{d_{01}d_{02}} + \frac{Y(X_1)}{d_{10}d_{12}} + \frac{Y(X_2)}{d_{20}d_{21}}$$

The independent variable value is the same as in Case 1.

$$X^* = \frac{-a_1}{2a_2}$$

<u>Case 4</u>

A cubic polynomial is fitted to $Y(X_o)$, $Y(X_1)$, $Y(X_2)$, $Y(X_3)$. The formulae for the polynomial coefficients are as follows

$$Y_i = Y(X_i)$$

$$B_{ij} = X_i X_j$$

$$d_{ij} = X_i - X_j$$

$$A_3 = -\frac{Y_o}{d_{10}d_{20}d_{30}} + \frac{Y_1}{d_{10}d_{21}d_{31}} - \frac{Y_2}{d_{20}d_{21}d_{32}} + \frac{Y_3}{d_{30}d_{31}d_{32}}$$

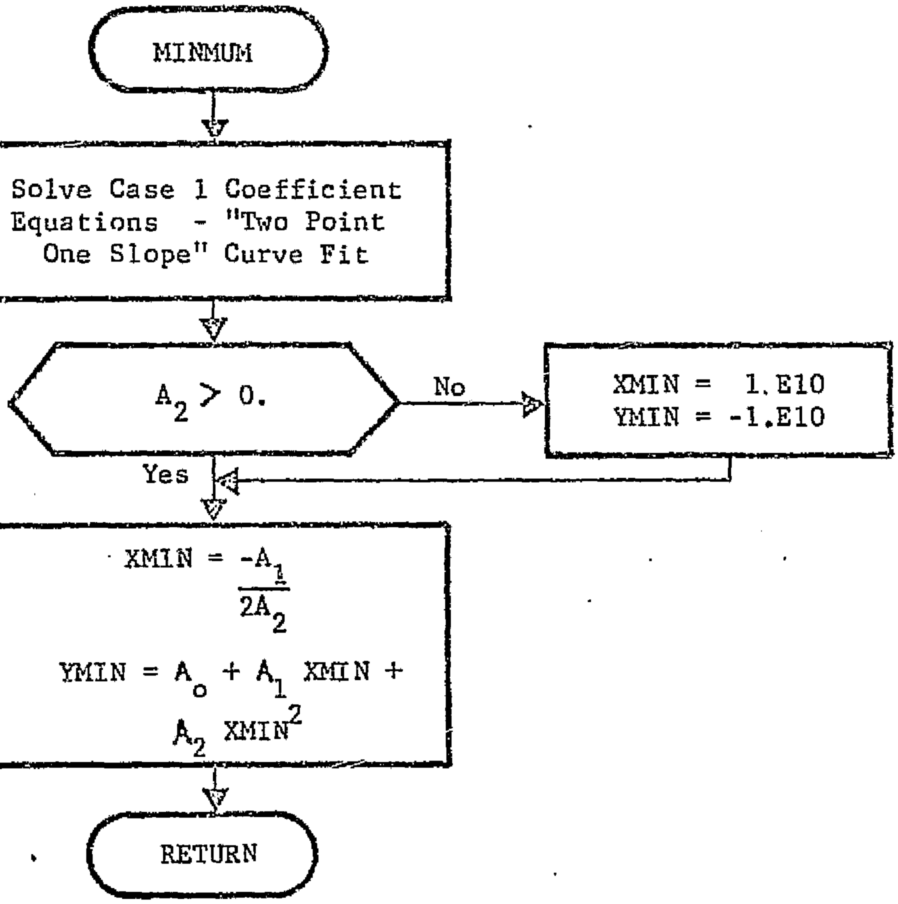
$$A_2 = \frac{(X_1+X_2+X_3)}{d_{10}d_{20}d_{30}} - Y_a - \frac{(X_o+X_2+X_3)}{d_{10}d_{21}d_{31}} - Y_1 + \frac{(X_o+X_1+X_3)}{d_{20}d_{21}d_{32}} \cdot Y_2 - \frac{(X_o+X_1+X_2)}{d_{30}d_{31}d_{32}} Y_3$$

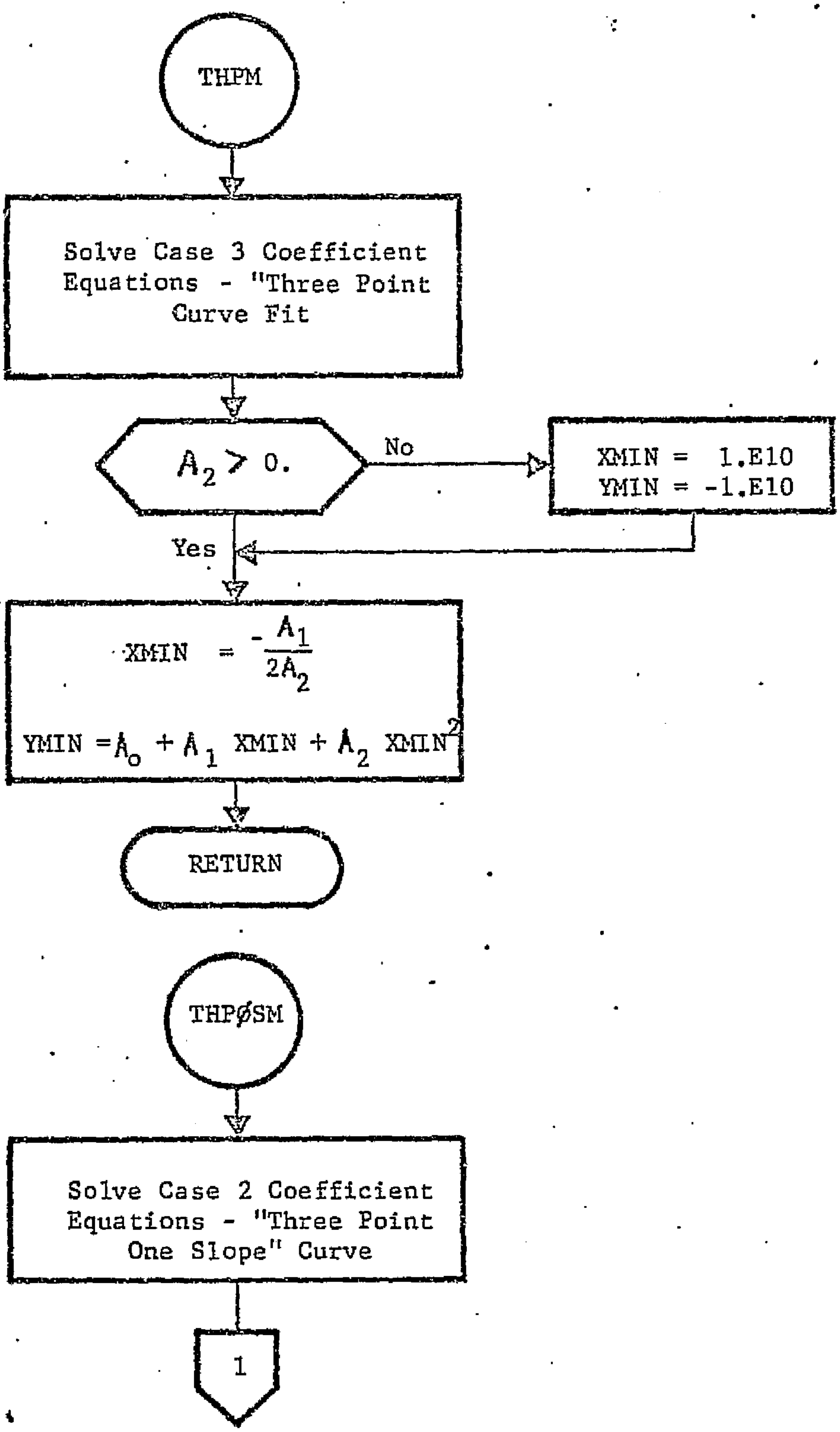
$$A_1 = -\frac{(B_{31}+B_{31}+B_{32})}{d_{10}d_{20}d_{30}} \cdot Y_o + \frac{(B_{20}+B_{30}+B_{32})}{d_{10}d_{21}d_{31}} Y_1 - \frac{(B_{10}+B_{30}+B_{31})}{d_{20}d_{21}d_{32}} + \frac{(B_{10}+B_{20}+B_{21})}{d_{30}d_{31}d_{32}} Y_3$$

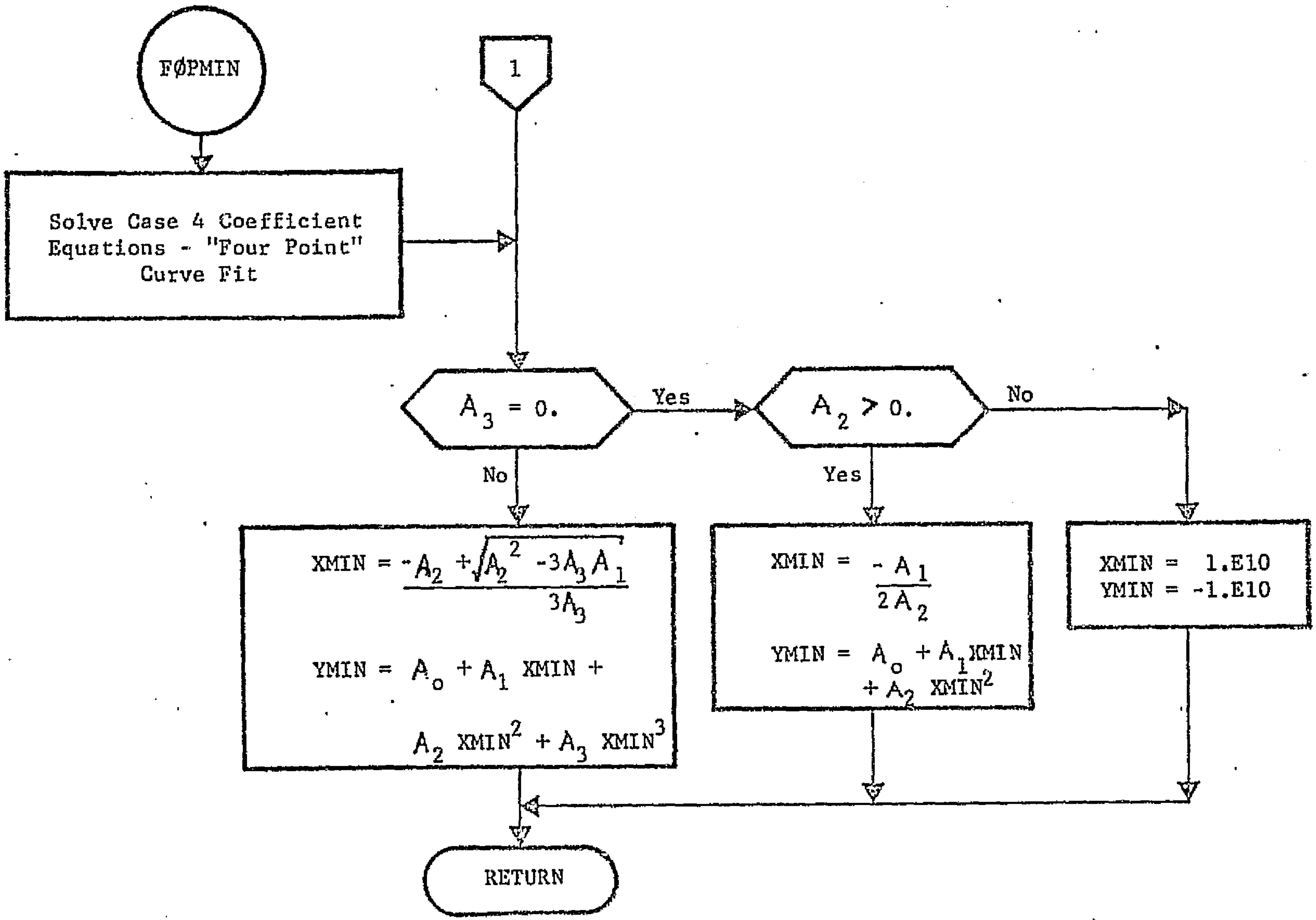
$$A_o = Y_o - (A_1X_o + A_2X_o{}^2 + A_3X_o{}^3)$$

The independent variable value minimizing P is the same as that in Case 2:

$$X^* = \left[-A_2 + \sqrt{A_2{}^2 - 3A_3A_1} \right] \cdot (3A_3)^{-1}$$


MINMUM
Solve Case 1 Coefficient Equations - "Two Point One Slope" Curve Fit
$A_2 > 0.$
No
XMIN = 1.E10
YMIN = -1.E10
Yes
$XMIN = \frac{-A_1}{2A_2}$
$YMIN = A_o + A_1\ XMIN + A_2\ XMIN^2$
RETURN


THPM
Solve Case 3 Coefficient Equations - "Three Point Curve Fit
$A_2 > 0.$
No
XMIN = 1.E10
YMIN = -1.E10
Yes
$XMIN = -\frac{A_1}{2A_2}$
$YMIN = A_o + A_1\ XMIN + A_2\ XMIN^2$
RETURN
THPØSM
Solve Case 2 Coefficient Equations - "Three Point One Slope" Curve
1


FØPMIN
1
Solve Case 4 Coefficient Equations - "Four Point" Curve Fit
$A_3 = 0.$
Yes
No
$A_2 > 0.$
Yes
No
$XMIN = \frac{-A_2 + \sqrt{A_2^2 - 3A_3A_1}}{3A_3}$
$YMIN = A_o + A_1\ XMIN + A_2\ XMIN^2 + A_3\ XMIN^3$
$XMIN = \frac{-A_1}{2A_2}$
$YMIN = A_o + A_1 XMIN + A_2\ XMIN^2$
XMIN = 1.E10
YMIN = -1.E10
RETURN

3.2.9 Subroutine: PGM

Purpose: To generate a targeted and optimized reference trajectory.

Method: PGM (Projected Gradient Method) is the organizational routine for the targeting and optimization submode. The logic for a complete iteration may be found in this routine. Basically, the iterative scheme proceeds as follows:

- o A reference trajectory is generated using the namelist input variables in $TRAJ.
- o The target error index is calculated.
- o The method of control correction is determined and convergence is tested.
- o Target sensitivities to changes in controls are computed by numerical differencing or STM techniques.
- o A control correction is computed and scaled.
- o The control correction is applied to the current control vector.
- o The trajectory associated with the new control vector becomes the reference trajectory for the next iteration.

This process continues until convergence has been achieved or the maximum number of iterations has been reached.

Remarks: A check is made on the remaining central processor, (CP), time after every iteration. If the estimated processor time for the next iteration is larger than the remaining CP time, the iteration process is terminated.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
CHI	O	C	In plane ΔV direction angle at injection.
DELVO	O	C	Injection ΔV.
E	I	C	Target errors evaluated for the current trajectory.
EMAG	I	C	Target error index.
EPSØN	I	C	Scalar multiple for control perturbations.
ETR (I,1)	O	C	I = 1, NT; Target errors of the reference trajectory for the current iteration.
F	I	C	Performance index of the current trajectory.
FTR(1)	O	C	Performance index of the reference trajectory for the current iteration.
GAMA	I	C	Scale factor providing the best control change.

Variable	Input/ Output	Argument/ Common	Definition
H	I/O	C	Control perturbation array.
INJLØC	I	C	Index on the control preceding the injection controls in the vector U.
INSG	I/O	C	Flag set when S and G are not calculated for current iteration.
ITERAT	O	C	Iteration counter.
KMAX	I	C	Number of thrust controls (THRUST(I,J)) chosen to be mode controls (U).
KØNVRJ	I	C	Convergence flag.
LØCDU	I	C	Blank common location of the total control correction vector (not scaled by GAMA).
LØCDU1	I	C	Blank common location of the performance control correction vector (not scaled by GAMA).
LØCDU2	I	C	Blank common location of the constraint control correction vector (not scaled by GAMA).
LØCRFM	I	C	Blank common location of the S/C masses evaluated at event times for the reference and all trial trajectories in a single iteration.
LØCSI*	I	C	Blank common location of the pseudo inverse of the weighted sensitivity matrix.

Variable	Input/ Output	Argument/ Common	Definition
LØCTS	I	C	Blank common location of event times for the reference and all trial trajectories in a single iteration.
LØCUL	I	C	Blank common location of minimum and maximum control bounds (ULIMIT).
LØCWG*	I	C	Blank common location of the weighted performance gradient.
LØCWS*	I	C	Blank common location of the weighted sensitivity matrix.
LØCWU	I	C	Blank common location of the control weights.
LØCXR	I	C	Blank common location of the 6-component state vectors associated with the event times of the reference and all the trail trajectories evaluated during a single iteration.
MIN	I	C	Index of the scale factor in the GAMMA vector which provides the best control correction.
NLP	I	C	Integer designation of the launch planet.
NT	I	C	Number of targets.
NTNP	I	C	Vector of primary bodies associated with the event times of the reference and all trial trajectories in a single iteration.

*Arrays may be in compressed form if controls have been dropped during the iteration.

Variable	Input/ Output	Argument/ Common	Definition
NTR	I	C	Trial Trajectory counter.
NU	I	C	Number of controls.
PMASS	I	C	Vector of planetary gravitational constants.
PRTURB	I	C	Vector of control perturbations.
PSI	Ø	C	Out of plane ΔV direction angle at injection.
STATEO	I/Ø	C	S/C state at trajectory start time for the reference trajectory of a given iteration.
STATR	I/Ø	C	Array of initial S/C states for the reference and all trial trajectories of a given iteration.
TARNØM	I/Ø	C	Target values evaluated for the reference trajectory.
TARTR	I	C	Target values evaluated for the reference trajectory and all trial trajectories in a given iteration.
U	I/Ø	C	Selection of controls for the specified mode run.
VPARK	Ø	C	Parking orbit velocity at injection.
WE	I	C	Vector of target weights.
XMM	Ø	C	Mean motion of S/C in parking orbit.
IASTM	I	C	Flag specifying method of computing the targeting sensitivity matrix.
IMØDE	Ø	C	TOPSEP submode flag.

Variable	Input/ Output	Argument/ Common	Definition
PRO	I	C	Radial distance of S/C at injection.
PINC	I	C	Ecliptic inclination of S/C at injection.
PTO	I	C	Injection time relative to launch epoch.
TUG	I	C	Logical flag specifying injection computations (TRUE).

Local Variables:

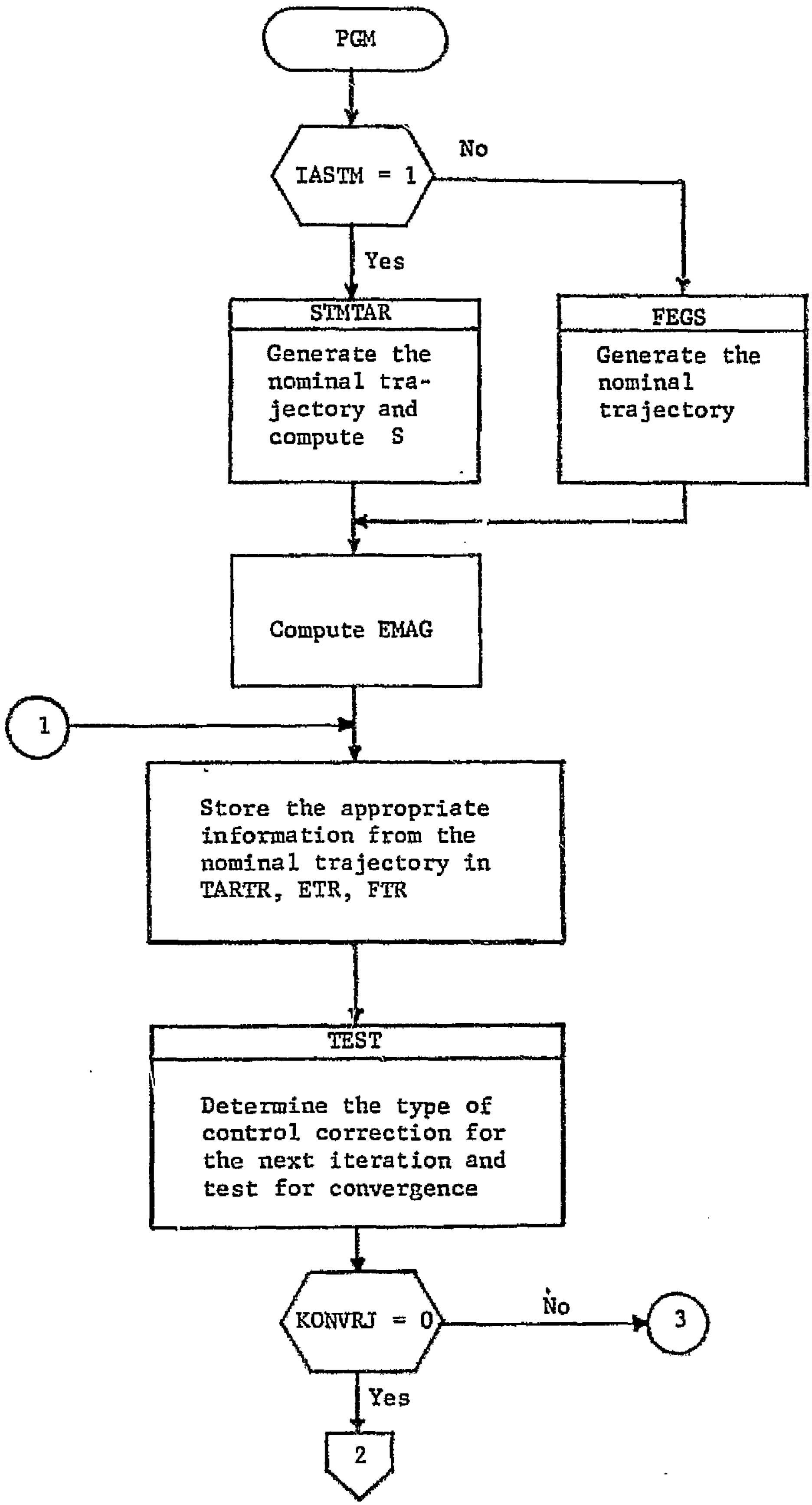
Variable	Definition
KØUNT	Index counter for the control vector U.
TCPITR	CP time for the first iteration (excluding reference trajectory generation).
TCPNØW	Current CP time relative to the start of the job.
TCPREF	CP time from job start to the end of the reference trajectory generation.

Subroutines Called: CØPY, FEGS, MMATBA, PRINTO, SECØND, SIZE, STEP, TEST, TIMELIM, ZERØM, STMTAR, INJECT

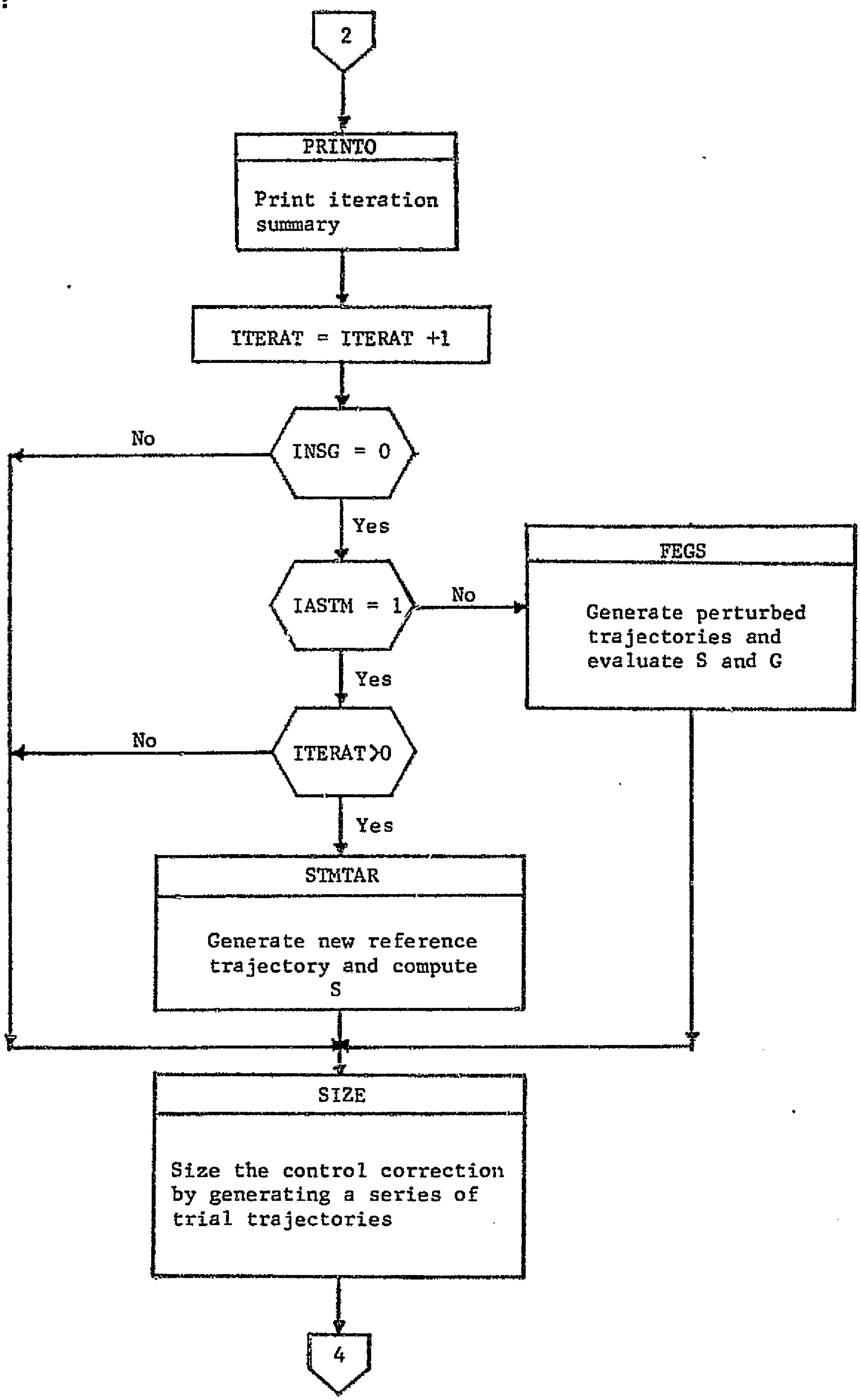
Calling Subroutines: TØPSEP

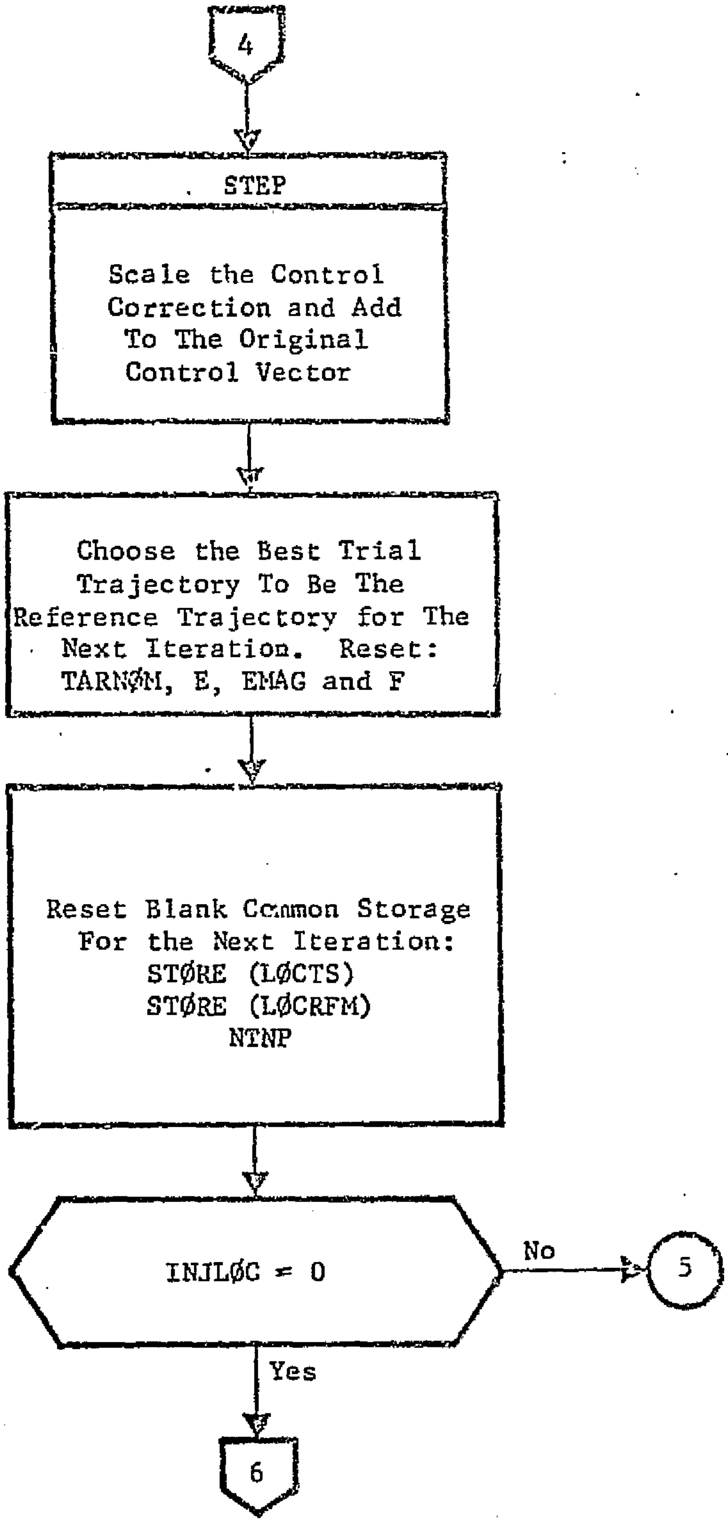
Common Blocks: (BLANK), CØNST, EDIT, EPHEM, TØP1, TØP2, TRAJ1, TRAJ2, WØRK, IASTM, TUG

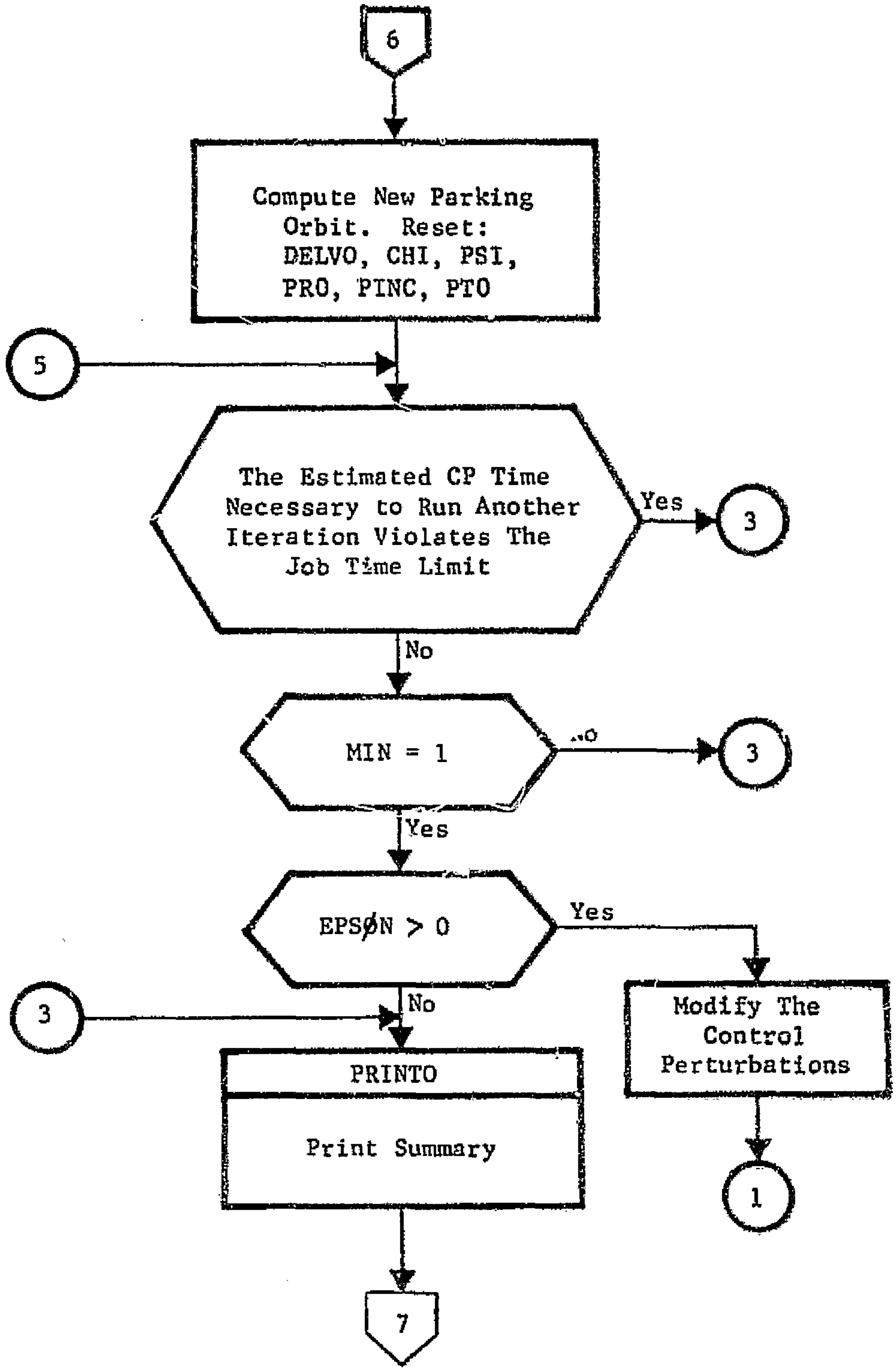
Logic Flow:

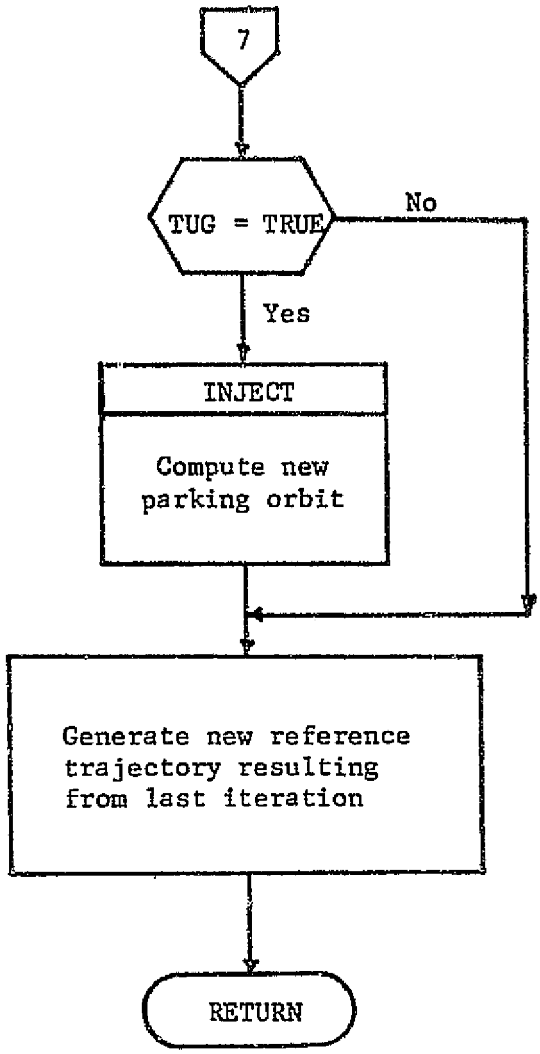

PGM
IASTM = 1
No
Yes
STMTAR
Generate the nominal trajectory and compute S
FEGS
Generate the nominal trajectory
Compute EMAG
1
Store the appropriate information from the nominal trajectory in TARTR, ETR, FTR
TEST
Determine the type of control correction for the next iteration and test for convergence
KONVRJ = 0
No
3
Yes
2

Logic Flow:


2
PRINTO
Print iteration summary
ITERAT = ITERAT +1
INSG = 0
No
Yes
IASTM = 1
No
FEGS
Generate perturbed trajectories and evaluate S and G
Yes
ITERAT>0
No
Yes
STMTAR
Generate new reference trajectory and compute S
SIZE
Size the control correction by generating a series of trial trajectories
4


4
STEP
Scale the Control Correction and Add To The Original Control Vector
Choose the Best Trial Trajectory To Be The Reference Trajectory for The Next Iteration. Reset: TARNØM, E, EMAG and F
Reset Blank Common Storage For the Next Iteration:
STØRE (LØCTS)
STØRE (LØCRFM)
NTNP
INJLØC = 0
No
5
Yes
6


6
Compute New Parking Orbit. Reset: DELVO, CHI, PSI, PRO, PINC, PTO
5
The Estimated CP Time Necessary to Run Another Iteration Violates The Job Time Limit
Yes
3
No
MIN = 1
3
Yes
EPSØN > 0
Yes
3
No
PRINTO
Print Summary
Modify The Control Perturbations
1
7


7
TUG = TRUE
No
Yes
INJECT
Compute new parking orbit
Generate new reference trajectory resulting from last iteration
RETURN

3.2.10 Subroutine: PRINTO (KFLAG)

Entry Points: PRINT1, PRINT2, PRINT3

Purpose: To provide print summaries for the various TOPSEP submodes.

Remarks: An iteration summary, a perturbed trajectory summary, a grid summary, or a termination summary is printed depending upon the entry point called.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
CNVRTT	I	C	Target parameter conversion constants.
CNVRTU	I	C	Control parameter conversion constants.
DPSI	I	C	Target error to be removed during current iteration.
DP2	I	C	Region of linearity in control space.
E	I	C	Target errors.
EMAG	I	C	Target error index.
ETØL	I	C	Target tolerances.
ETR	I	C	Array of target errors for iteration trial steps.
F	I	C	Performance index.
FTR	I	C	Vector of performance indices for iteration trial steps.
G	I	C	Performance gradient.
GAMA	I	C	Optimum control change scale factor.

Variable	Input/ Output	Argument/ Common	Definition
ITERAT	I	C	Iteration number.
KFLAG	I	A	Print specification flag.
KØNVRJ	I	C	Convergence flag.
KØUNT	I	C	Index on control under consideration.
LABELT	I	C	Hollerith target labels.
LØCDU	I	C	Blank common location of total control correction vector.
LØCDU1	I	C	Blank common location of performance control correction vector.
LØCDU2	I	C	Blank common location of the targeting control correction vector.
LØCEM1	I	C	Blank common location of the target error indices associated with the first step of the control grid.
LØCEM2	I	C	Blank common location of the target error indices associated with the second step of the control grid.
LØCEN	I	C	Blank common location of the target errors associated with the first step of the control grid.
LØCE2	I	C	Blank common location of the target errors associated with the second step of the control grid.
LØCF1	I	C	Blank common location of the performance indices associated with the first step of the control grid.

Variable	Input/ Output	Argument/ Common	Definition
LØCF2	I	C	Blank common location of the performance indices associated with the second step of the control grid.
NT	I	C	Number of targets.
NU	I	C	Number of controls.
PG2	I	C	The square of the projected gradient magnitude.
PRTURB	I	C	Control perturbation.
S	I	C	The sensitivity matrix.
TARGET	I	C	Desired target values.
TARPAR	I	C	Target values of perturbed trajectories.
TARTR	I	C	Target values of the trial trajectories.
U	I	C	Control vector.
LABEL	I	C	Hollerith labels for all possible targets.
XINC	I	C	Ecliptic inclination.
ØMEGA	I	C	Longitude of ascending mode.
SØMEGA	I	C	Argument of periapsis.
XMEAN	I	C	Mean anomaly.
TA	I	C	True anomaly.

Local Variables:

Variable	Definition
CDU (= WORK(121))	The scaled control change (converted to output units).
DU1ØUT (= WØRK(1))	Converted performance control change.
DU2ØUT (= WØRK(21))	Converted constraint control change.

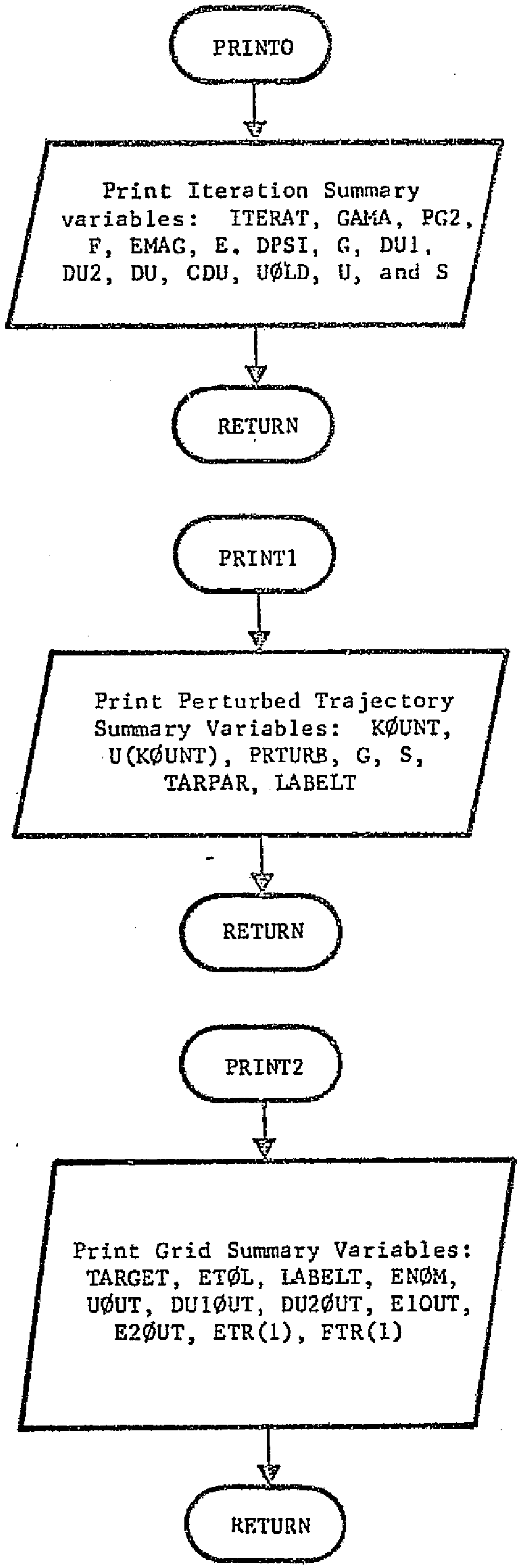
Variable	Definition
ENØM (= WØRK(73))	Converted target errors of the nominal trajectory.
ETLØUT (= WØRK(85))	Converted target tolerances.
E1ØUT (= WØRK(61))	Converted target errors of the first step grid trajectories.
E2ØUT (= WORK(67))	Converted target errors of the second step grid trajectories.
TARØUT (= WORK(79))	Converted target values.
UØLD (= WORK(101))	Converted control vector of previous iteration.
UØUT (= WØRK(41))	Converted control vector.
WØRK	Working storage.
ISTØPN	Hollerith labels of requested stopping conditions.
KØFF	Hollerith labels of actual stopping conditions.

Subroutines Called: SCALE, STEP

Calling Subroutines: FEGS, GRID, PGM, TREK, STMTAR

Common Blocks: (BLANK), GRID, PRINTH, TØP1, TØP2, WØRK, TARGET

Logic Flow:


PRINTO
Print Iteration Summary variables: ITERAT, GAMA, PG2, F, EMAG, E. DPSI, G, DU1, DU2, DU, CDU, UØLD, U, and S
RETURN
PRINT1
Print Perturbed Trajectory Summary Variables: KØUNT, U(KØUNT), PRTURB, G, S, TARPAR, LABELT
RETURN
PRINT2
Print Grid Summary Variables: TARGET, ETØL, LABELT, ENØM, UØUT, DU1ØUT, DU2ØUT, E1OUT, E2ØUT, ETR(1), FTR(1)
RETURN

3.2.11 Subroutine: PRINTD

Purpose: To print submode input summaries.

Remarks: PRINTD is in the DATAT overlay and does not remain in core during TOPSEP's submode operation.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
CNTRØL	I	C	Initial values of all possible controls.
CNVRTU	I	C	Conversion constants from input units to internal units for selected controls.
DFMAX	I	C	Maximum increase allowed in the cost index (F).
DP2	I	C	Estimated region of linearity in the control space.
EPSØN	I	C	Scalar multiple for control perturbations.
GØUT	I	C	Performance gradient in print-out units.
GTRIAL	I	C	One-dimensional search constants.
HØUT	I	C	Control perturbations in printout units.
IMØDE	I	C	TOPSEP submode designation.
INACTV	I	C	Vector denoting which controls are active, on bounds, or within bound tolerance regions.
INSG	I	C	Flag set to 1 when S and G are input through namelist (nominally 0).

Variable	Input/ Output	Argument/ Common	Definition
IWATE	I	C	Flag designating the desired control weighting scheme.
JMAX	I	C	Number of mission thrust phases.
KMAX	I	C	Number of thrust controls (THRUST (I, J)) chosen to be elements in U.
KNTRØL	I	C	Hollerith names for the elements in CONTRØL.
LØCUL	I	C	Blank common location of minimum and maximum control bounds.
NMAX	I	C	Maximum number of iterations.
NT	I	C	Number of targets.
NU	I	C	Number of controls.
PCT	I	C	Percentage of target error to be removed during an iteration.
SØUT	I	C	Target sensitivity matrix in printout units.
STØL	I	C	Test variable for determining linearly dependent columns of the weighted sensitivity matrix.
TLØW	I	C	Limit of target error index below which optimization only is performed.
TUP	I	C	Limit of target error index above which simultaneous targeting and optimization is discontinued and targeting only is initiated.

Variable	Input/ Output	Argument/ Common	Definition
UWATE	I/O	C	User input control weights.
WØRK	I	C	Working storage.

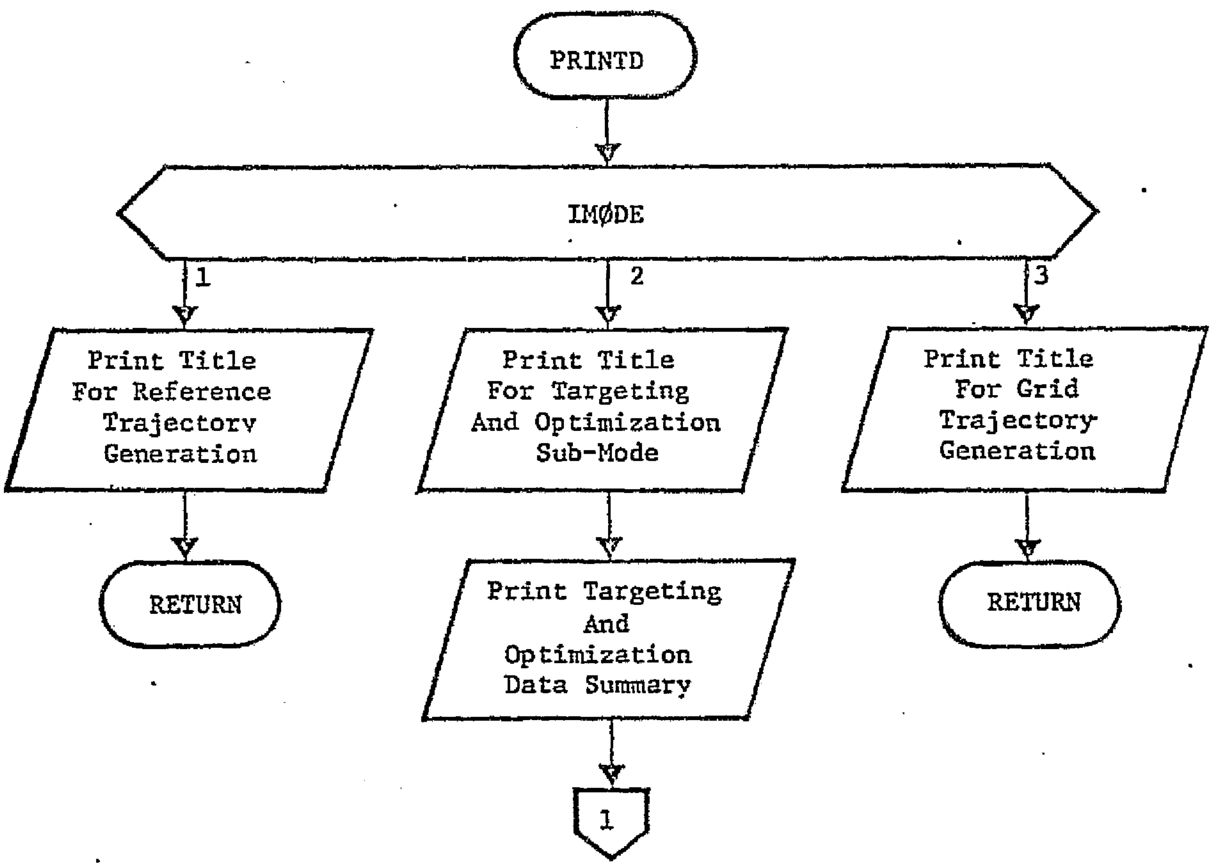
Local Variables:

Variable	Definition
KØUNT	Control counter.
UL (= WØRK(1))	The minimum and maximum values of the control bounds in printout units.

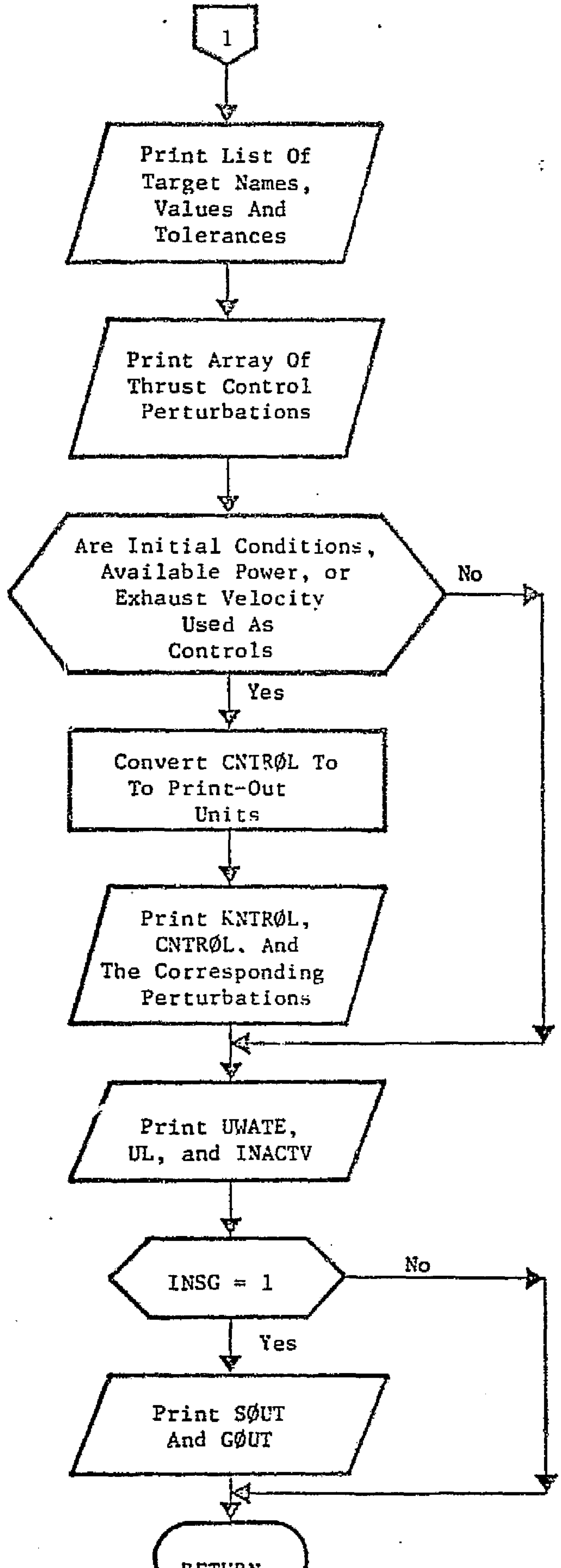
Subroutines Called: None

Calling Subroutines: DATAT

Common Blocks: (BLANK), CØNST, EDIT, EPHEM, GRID, PRINT, PRINTH, TIME, TØP1, TØP2, TRAJ1, TRAJ2, WØRK



Logic Flow:


1
Print List Of Target Names, Values And Tolerances
Print Array Of Thrust Control Perturbations
Are Initial Conditions, Available Power, or Exhaust Velocity Used As Controls
No
Yes
Convert CNTRØL To To Print-Out Units
Print KNTRØL, CNTRØL, And The Corresponding Perturbations
Print UWATE, UL, and INACTV
INSG = 1
No
Yes
Print SØUT And GØUT

3.2.12 Subroutine: SIZE

Purpose: To size the control correction.

Method: The basic procedure for sizing the control correction is as follows:

1. Compute the target error to be removed during the current iteration. Often it is not wise to remove all the target error in one step due to the nonlinear relationship of the targets to the controls.
2. Compute the control correction $\Delta \underline{U}$ based upon the method of projected gradients.
3. Perform a one-dimensional search in the $\Delta \underline{U}$ direction to determine a scaled control correction which will minimize either the target error, the cost index, or both.

Supplementary computations include:

- o Determining linear dependency among columns of the sensitivity matrix, S, thus averting numerical problems when computing the pseudo-inverse of S.
- o Determining which controls lie on their respective bounds and which control corrections violate the control constraints.
- o Determining the maximum allowable scale factor for the current iteration

<u>Remarks:</u> Steps 1 and 2 of the control sizing procedure are completed in the secondary overlay DELTU which is called from SIZE. In addition, DELTU performs most of the supplementary calculations. The third step is completed within subroutine GENMIN. Subroutine SIZE monitors the overall procedure. Elaboration of the third step in terms of the coded logic follows.

Subroutine size calls subroutine GENMIN to compute the value of the scaling factor γ (GAMA) which minimizes a function $P(\gamma)$ in the combined constraint direction, $\Delta \underline{u}_2$, and the optimization direction, $\Delta \underline{u}_1$, or each direction individually depending upon the value of NTYPE. The function $P(\gamma)$ is the sum of two functions, $P1(\gamma)$ and $P2(\gamma)$. $P1(\gamma)$ is the net cost index and $P2(\gamma)$ is the target error index.

$$P(\gamma) = \alpha \cdot \lambda \cdot P1(\gamma) + \beta \cdot P2(\gamma)$$

where

$$\alpha = \begin{cases} 1, & \text{for optimization only or simultaneous targeting and optimization,} \\ 0, & \text{for targeting only} \end{cases}$$

$$\beta = \begin{cases} 1, & \text{for targeting only or simultaneous targeting and optimization,} \\ 0, & \text{for optimization only} \end{cases}$$

λ = Weighting of the net cost index (ØSCALE)

GENMIN evaluates $P(\gamma)$ for different values of γ so that a polynomial approximation of the function can be made. Once the polynomial is formulated the minimizing γ may be computed analytically. To reduce the number of point evaluations of $P(\gamma)$, SIZE provides GENMIN with the first derivative of the function at $\gamma = 0$. The first derivative (DP12DS) is of the form

$$P'(0) = \left. \frac{d\,P(\gamma)}{d\gamma} \right|_{\gamma = 0} = \alpha \cdot \lambda \cdot P1'(0) + \beta \cdot P2'(0)$$

For the special case when only the target error is to be minimized, the first derivative (DP2DS) is

$$P'(0) = P2'(0)$$

Likewise, for the case when only the net cost is to be minimized, the first derivative (DP1DS) is

$$P'(0) = \lambda \cdot P1\ (0)$$

The function $P2(\gamma)$ to be minimized along the constraint direction, Δu_2, is the sum of the squares of the target errors ($\underline{E}$) divided by the target tolerances (ETØL).

$$P2(\gamma) = \underline{E}^T(\underline{u} + \gamma \Delta \underline{u}_2)\ W\ \underline{E}(\underline{u} + \gamma \Delta \underline{u}_2)$$

where

$$W = \begin{bmatrix} \left(\frac{1}{ET\emptyset L(1)}\right)^2 & & & 0 \\ & \left(\frac{1}{ET\emptyset L(2)}\right)^2 & & \\ & & \ddots & \\ 0 & & & \left(\frac{1}{ET\emptyset L(NT)}\right)^2 \end{bmatrix}$$

The first derivative evaluated at $\gamma = 0$ is simply

$$P2'(0) = 2\underline{E}^T(\underline{u})\, S\, \Delta \underline{u}_2$$

where S is the target sensitivity matrix $(\frac{\delta \underline{\underline{E}}}{\delta \underline{u}})$.

The function $P1(\gamma)$ to be minimized along the optimization direction $\Delta \underline{u}_1$ is defined

$$P1(\gamma) = \overbrace{F(\underline{u} + \gamma \Delta \underline{u}_1) - F(\underline{u})}^{A} + \underbrace{G^T(\underline{u}) \left[-S(SS^T)^{-1} E(\underline{u} + \gamma \Delta \underline{u}_1) \right]}_{B}$$

where A represents the change in performance produced by a step of length γ along $\Delta \underline{u}_1$ and B represents the linearized approximation to change in performance required to eliminate the target error produced by a step of length γ along $\Delta \underline{u}_1$. F is the cost index (negative of the S/C mass) and G is the cost gradient $(\frac{\partial F}{\partial \underline{u}})$. The first derivative evaluated at $\gamma = 0$ is then

$$P1'(0) = \underline{G}^T (\underline{u}) \; \Delta \underline{u}_1$$

The functions $P'(0)$, $P1'(0)$, and $P2'(0)$ are initialized in the secondary overlay DELTU. The point evaluations of the functions $P(\gamma)$, $P1(\gamma)$, and $P2(\gamma)$ are computed in GENMIN and stored in the vectors P1P2, P1, and P2 respectively. The various values of the scale factor, γ, are stored in the vector GAMMA while the minimizing scale factor is stored in the variable GAMA.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
BIG	I	C	Large constant, 1.E20
DP1DS	I	C	P1'(0)
DP12DS	I	C	P'(0)
DP2DS	I	C	P2' (0)
DP2	I/O	C	Scale on optimization correction.
GAMA	O	C	Scale factor providing the best control change.
GAMMA	O	C	Vector of control change scale factors for the trial trajectories.
GMAX	O	C	Largest allowed scale factor.
GTRIAL	I/O	C	One-dimensional search constants.
INACTV	I/O	C	Vector denoting which controls are active (1), on bounds (0), or within bound tolerances.
INSG	I/O	C	Flag set when S and G are input through namelist.
ITERAT	I	C	Iteration counter.
KGMAX	I	C	Index on control which will reach a bound if GMAX scales Δu.
LØCUL	I	C	Blank common location for the control bounds.
MIN	O	C	Index of minimizing scale factor in GAMMA.
NTYPE	O	C	Flag specifying the type of control correction.

Variable	Input/ Output	Argument/ Common	Definition
NU	I	C	Number of controls.
P1	O	C	Vector of net cost values corresponding to the scale factors in GAMMA.
P1P2	O	C	Vector of combined net cost and target error index values corresponding to the scale factors in GAMMA.
P2	O	C	Vector of target error index values corresponding to the scale factors in GAMMA.
U	I	C	Control vector.
UL1M1T	I	C	Control bounds.

Local Variables:

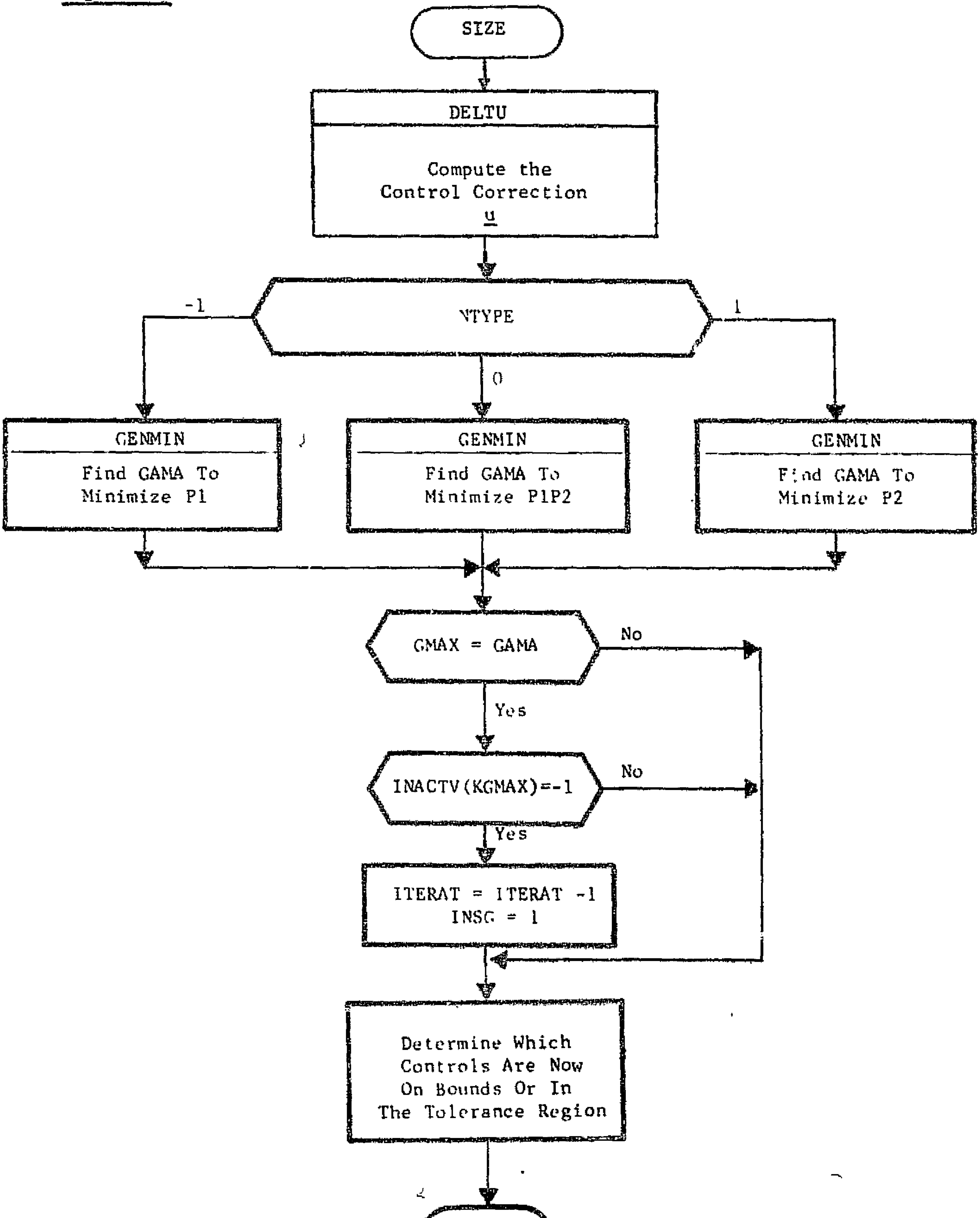
Variable	Definition
P1EST	Vector containing the estimates of P1(γ) for the trial trajectories.
P12EST	Vector containing the estimates of P(γ) for the trial trajectories.
P2EST	Vector containing the estimates of P2(γ) for the trial trajectories.
UNEW	Updated control vector used to compute INACTV.

Subroutines Called: CØPY, DELTU, GENMIN, STEP

Calling Subroutines: PGM

Common Blocks: (BLANK), CØNST, EDIT, TØP1, TØP2, WØRK, SIZE*

Logic Flow:


SIZE
DELTU
Compute the Control Correction u
NTYPE
-1
0
1
GENMIN
Find GAMA To Minimize P1
GENMIN
Find GAMA To Minimize P1P2
GENMIN
Find GAMA To Minimize P2
GMAX = GAMA
No
Yes
INACTV(KGMAX)=-1
No
Yes
ITERAT = ITERAT -1
INSG = 1
Determine Which Controls Are Now On Bounds Or In The Tolerance Region

Pages 159 through 165 have been deleted.

3.2.13 Subroutine: STEP (UØLD, SCALE, DELU, NU, UNEW)

Purpose: To update the control vector.

Method: The new control vector is updated by the following algorithm:

UNEW (I) = UØLD (I) + SCALE * DELU (I)

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
DELU	I	A	Control correction vector.
NU	I	A	Number of controls.
SCALE	I	A	Scale on control correction.
UNEW	O	A	Updated control vector.
UØLD	I	A	Previous control vector.

Local Variables: None

Subroutines Called: None

Calling Subroutines: GRID, PGM

Common Blocks: None

3.2.14A **Subroutine:** STEST (WS, NT, NU, STØL, CDØTC, CMAG, LDEP, NDEP)

Purpose: To compute the inner products between columns of the weighted sensitivity matrix in order to determine linearly dependent control sensitivities.

Method: The normalized inner products between columns of the weighted sensitivity matrix are computed and stored in the CDØTC array. These values are then tested to determine whether they fall within some tolerance (STØL) of unity. The control sensitivity vectors, whose inner products do fall within this tolerance region, are considered to be linearly dependent and at least one of the associated controls will be dropped from the control vector during the concurrent iteration. For example, if $\underline{S}_i$ and $\underline{S}_j$ represent two columns of the weighted sensitivity matrix and

$$1 - \left| \frac{\underline{S}_i \cdot \underline{S}_j}{|\underline{S}_i| * |\underline{S}_j|} \right| < \text{STØL}$$

then $\underline{S}_i$ and $\underline{S}_j$ are considered linearly dependent. Whether the $\underline{u}_i$ and $\underline{u}_j$ component is dropped from the control vector depends upon the other column vector inner products. If $\underline{S}_j$ and $\underline{S}_k$ are also linearly dependent then control u_j will be dropped since this measure will allow more controls to remain active. The fact that a tolerance region is used to test linear dependency does permit $\underline{S}_i$ and $\underline{S}_k$ to remain linearly independent although both vectors are linearly dependent with $\underline{S}_j$. If $\underline{S}_i$ and $\underline{S}_j$ are the only linearly dependent vectors the control with the lower index is arbitrarily dropped.

Remarks: STEST is called only once per iteration and only when considering controls in the weighted space.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
CDØTC	O	A	Array of normalized inner products; CDØTC (I, J) is the inner product between the I and J columns of WS.
CMAG	O	A	Magnitude of the sensitivity column vectors.
LDEP	O	A	Vector of flags nominally zero but set to 1 to denote which controls should be dropped.
NDEP	O	A	Number of dropped controls.
NT	I	A	Number of targets.
NU	I	A	Number of controls.
STØL	I	A	Minimum difference allowed between normalized inner products of the control sensitivity vectors and unity before the vectors are considered linearly dependent.

Variable	Input/ Output	Argument/ Common	Definition
WS	I	A	Weighted sensitivity matrix.

Local Variables:

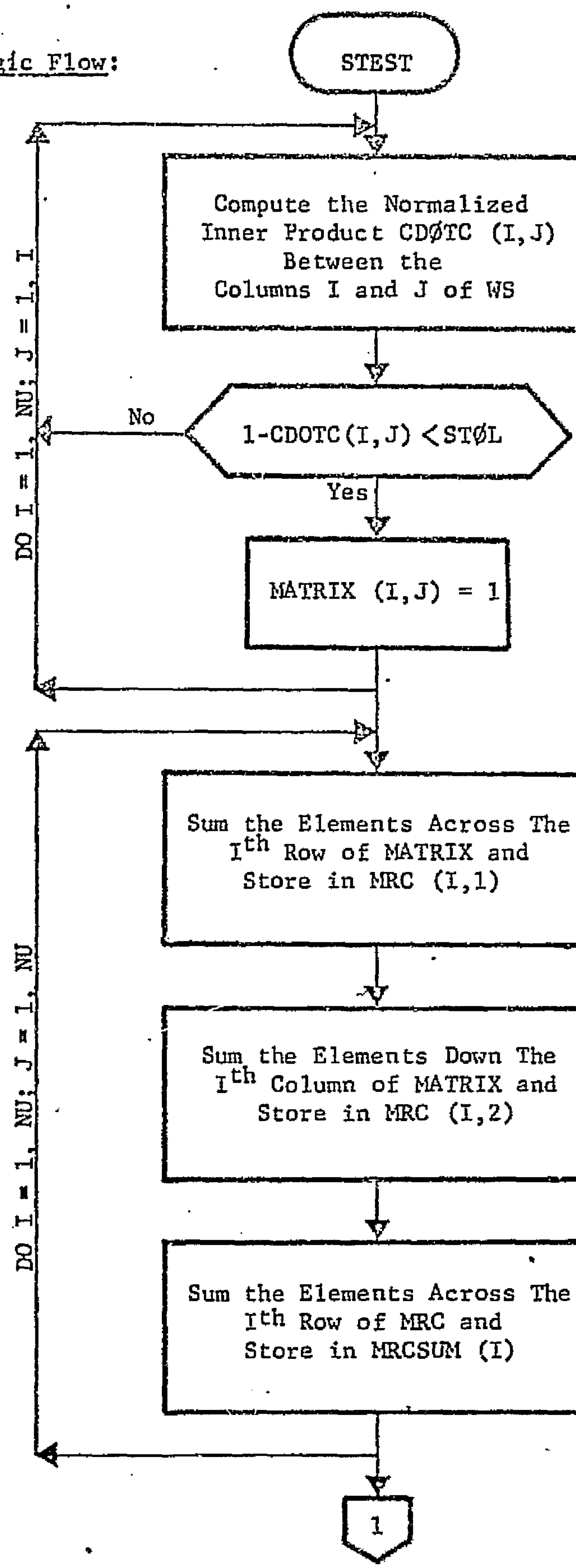
Variable	Definition
MATRIX	Integer array the same dimensions as CDOTC whose components are nominally zero but set to 1 when (1-CDØTC $_{ij}$ < STØL)
MRC	NU X 2 array; the first column represents the sum of the elements across the rows of MATRIX; the second column represents the sum of elements down the columns of MATRIX.
MRCSUM	NU X 1 vector whose elements represent the sum across the rows of MRC.
ITEST	Index of the largest element of MRCSUM.

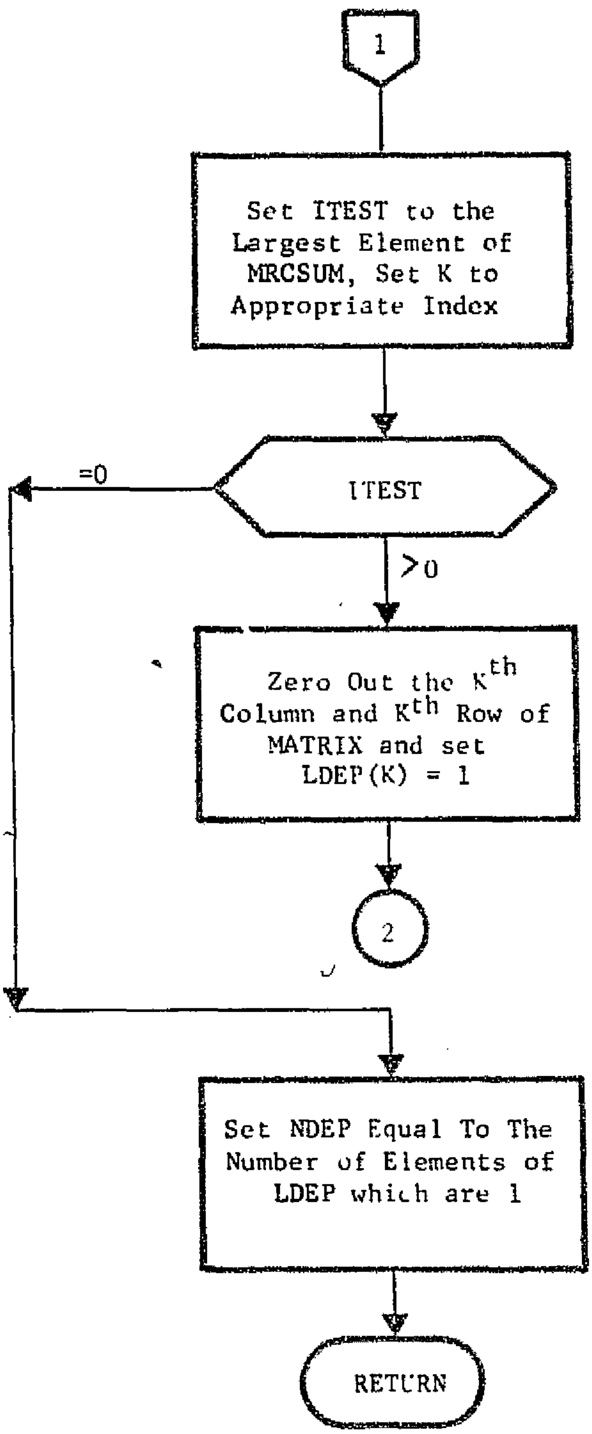
Subroutines Called: ZERØM

Calling Subroutines: SIZE

Common Blocks: None

Logic Flow:


STEST
Compute the Normalized Inner Product CDØTC (I,J) Between the Columns I and J of WS
1-CDOTC(I,J) <STØL
No
Yes
MATRIX (I,J) = 1
DO I = 1, NU; J = 1, I
Sum the Elements Across The I^th Row of MATRIX and Store in MRC (I,1)
Sum the Elements Down The I^th Column of MATRIX and Store in MRC (I,2)
Sum the Elements Across The I^th Row of MRC and Store in MRCSUM (I)
DO I = 1, NU; J = 1, NU
1


1
Set ITEST to the Largest Element of MRCSUM, Set K to Appropriate Index
ITEST
=0
>0
Zero Out the K^th Column and K^th Row of MATRIX and set LDEP(K) = 1
2
Set NDEP Equal To The Number of Elements of LDEP which are 1
RETURN

3.2.14B **Subroutine:** STMTAR (IT)

Purpose: To compute the targeting sensitivity matrix from the augmented state transition matrix.

Method: The method of computing the sensitivity matrix, S, from the partitions of the augmented STMs, Ø and Θ, is described in Reference 1, Section 9.7, page 140.

Remarks: During each iteration the reference trajectory (i.e. the trajectory defined by the $TRAJ variables in the zeroth iterate and the "best" trial trajectory in each subsequent iteration) must be integrated to compute Ø, Θ, and S. If a portion of this reference trajectory remains constant throughout the iterative process, it is integrated during the zeroth iterate only.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
CA	O	C	Closest approach computed in BPLANE
E	O	C	Target error vector
ETA(=STATR(1,2))	O	C	Sensitivity of targets to changes in final state
F	O	C	Cost index (negative of payload)
IJH	I	C	Array of flags indicating active controls
IPRINT	O	C	Trajectory print flag

Variable	Input/ Output	Argument/ Common	Definition
IT	I	A	Flag indicating integration of the fixed trajectory arc (-1) or integration of STMs (1)
KMAX	I	C	Number of active thrust controls
LISTAR	I	C	Array of flags indicating selected targets
LØCM	I	C	Blank common location of final S/C mass
LOCRFM	I	C	Blank common location of the S/C masses evaluated at event times
LOCTS	I	C	Blank common location of event times
LOCXR	I	C	Blank common location of the S/C states evaluated at event times
MPRINT	I	C	TOPSEP print flags
NPRI	I	C	Primary body designation
NT	I	C	Number of targets
NTNP	O	C	Vector of primary body designations associated with trajectory event times
NTP	I	C	The target body code
NTPH	I	C	Vector of control phase numbers associated with event times
NTPHAS	I	C	Thrust phase counter
NU	I	C	Number of controls
PHI	O	C	State transition matrix (6x6)
RCA	O	C	Target planet encounter radius computed in TRAJ
S	O	C	Targeting sensitivity matrix
SCMASS	I	C	S/C mass at trajectory start time

Variable	Input/ Output	Argument/ Common	Definition
STATEO	I	C	S/C state at trajectory start time
STATR	I	C	Array of initial states corresponding to the reference and each trial trajectory
TARGET	O	C	Desired target values
TARNØM	O	C	Target values evaluated for the reference trajectory
TCA	O	C	Time of closest approach computed in BPLANE
TEND	I	C	Trajectory end time
THETA	O	C	Sensitivity of final state to changes in thrust controls
TM	I	C	Time conversion constant (days to seconds)
TRCA	O	C	Time at closest approach computed in TRAJ
TSI	O	C	Time at SOI computed in BPLANE
TSØI	O	C	Time at SOI computed in TRAJ
TSTART	I	C	Trajectory start time
TUG	I/O	C	Logical flag indicating injection computations if TRUE

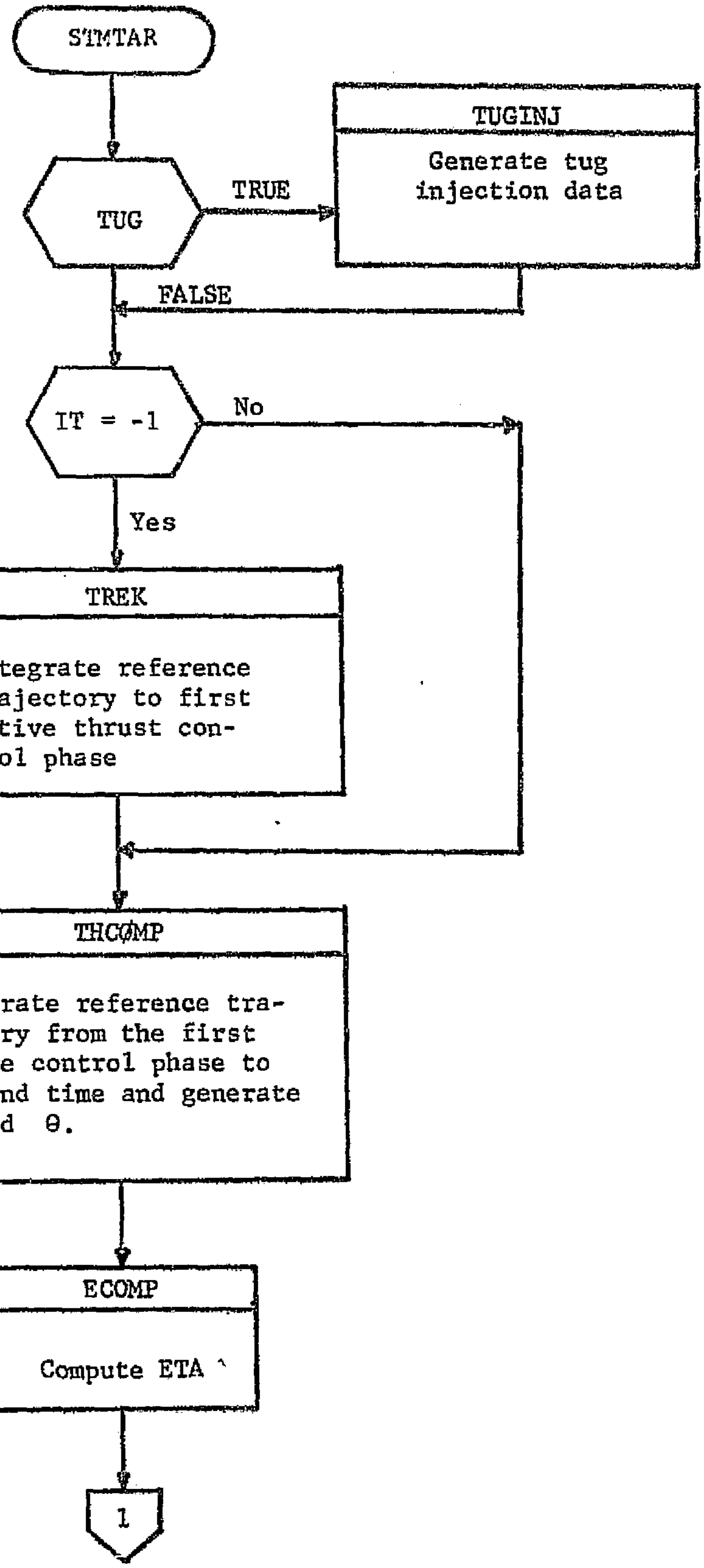
Local Variables:

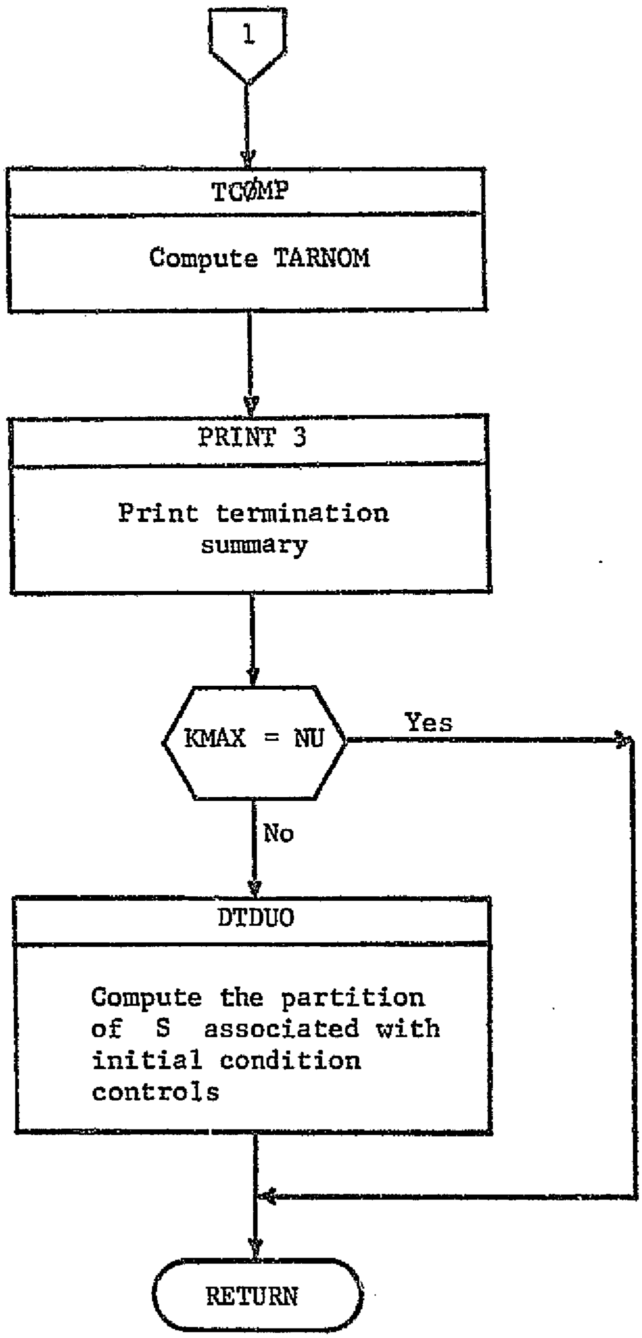
Variable	Definition
NPRIO	Primary body designation at time TSTART for the reference trajectory
REFMO	S/C initial mass at time TSTART for the reference trajectory
REFXO	S/C initial state at time TSTART for the reference trajectory

<u>Subroutines Called:</u> CØPY, DTDUO, ECØMP, MATØUT, MMAB, MUNPAK, PRINT3, SUB, TCØMP, THCOMP, TREK, TUGINJ, VECMAG

<u>Calling Subroutine:</u> PGM

<u>Common Blocks:</u> (Blank), CØNST, IASTM, TARGET, TIME TØPI, TØPZ, TRAJ1, TRAJ2, TUG


STMTAR
TUG
TRUE
TUGINJ
Generate tug injection data
FALSE
IT = -1
No
Yes
TREK
Integrate reference trajectory to first active thrust control phase
THCØMP
Integrate reference trajectory from the first active control phase to the end time and generate Ø and θ.
ECOMP
Compute ETA
1


1
TCØMP
Compute TARNOM
PRINT 3
Print termination summary
KMAX = NU
Yes
No
DTDUO
Compute the partition of S associated with initial condition controls
RETURN

3.2.15 Subroutine: TEST

Purpose: To test for convergence and to determine whether the next control change will be a targeting and/or optimization correction.

Method: The determination of the type of control correction is based upon the size of the error index (EMAG). The value of EMAG is compared to user input limits which direct the calculation of the next control change to be either a constraint correction, a performance correction, or simultaneous constraint and performance corrections. The iteration process is considered converged and the run is terminated when the performance index is maximized.

Remarks: A summary of the control correction decision process is given in the following table.

IF	THEN
EMAG > TUP	TARGETING
TLØW < EMAG < TUP	TARGETING AND OPTIMIZATION
EMAG < TLØW	OPTIMIZATION

Search Direction Options

The input limits TUP, TLØW, and ØPTEND allow the user flexibility in determining the type of targeting and optimization strategy. For example, the user may concentrate on targeting exclusively by setting TUP = TLØW = 1, and ØPTEND = 0. When the trajectory is targeted the run will terminate without optimizing.

The angle (θ) between $\underline{G}$ and $\Delta \underline{u}_1$ is used to test convergence in subroutine TEST. Optimization is considered complete when

$$\cos \theta = \frac{\underline{G} \cdot \Delta \underline{u}_1}{|\underline{G}| * |\Delta \underline{u}_1|}$$

approaches 0 (when θ approaches 90 deg) and when EMAG $<$ TLØW. The user may override this convergence requirement by specifying ØPTEND. When ØPTEND $< \theta <$ 90 and EMAG $<$ TLØW the run is terminated. Figure 3-1 illustrates the convergence process.

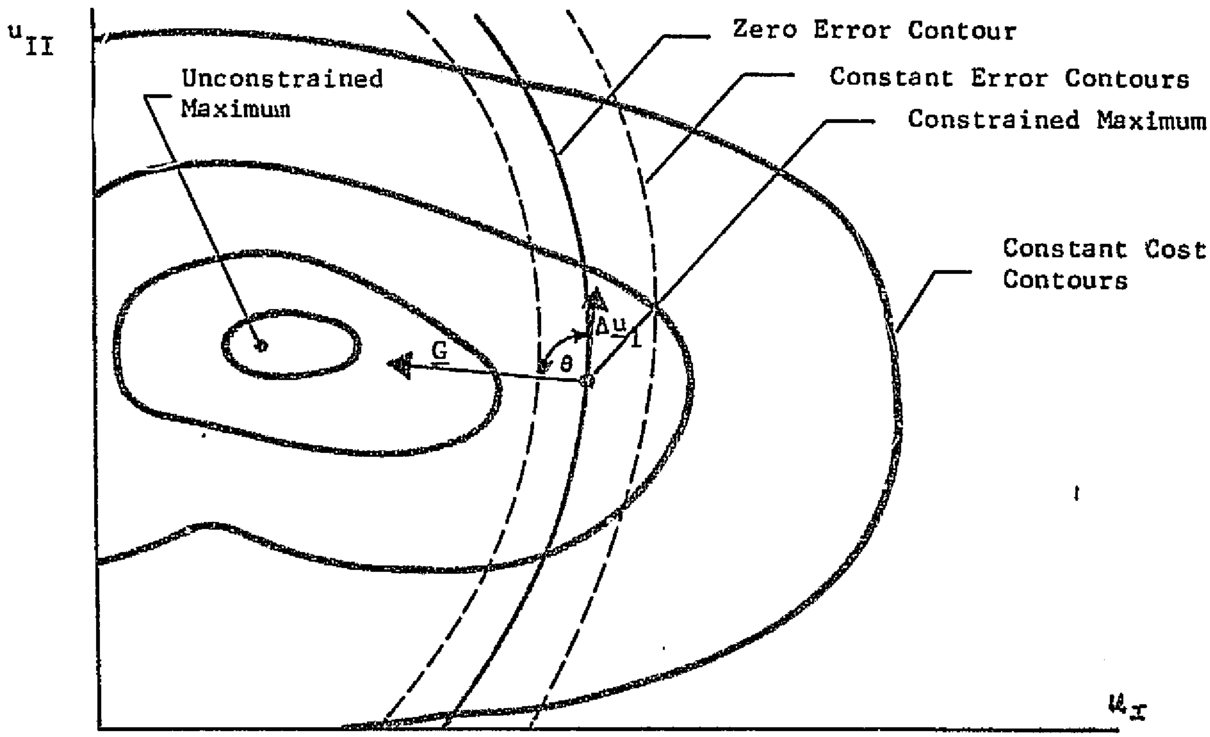


Figure 3-1 Geometric Interpretation of Convergence

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
CTHETA	I	C	Cosine of the convergence test angle, Θ. As optimization process converges, Θ approaches 90 degrees and CTHETA approaches 0.
EMAG	I	C	Quadratic error index.
ITERAT	I	C	Current iteration number.
KØNVRJ	O	C	Convergence flag.
			= -1, maximum iteration number reached
			= 0, iteration in process
			= 1, convergence
NMAX	I	C	Maximum number of iterations allowed.

Variable	Input/ Output	Argument/ Common	Definition
NTYPE	O	C	Flag designating type of next control correction. = -1, optimization only = 0, targeting and optimization = 1, targeting only
ØPTEND	I	C	User specified convergence tolerance on optimization process (e.g., CTHETA ≤ ØPTEND indicates convergence).
TLØW	I	C	Upper limit of EMAG for which optimization only is performed.
TUP	I	C	Lower limit of EMAG for which targeting only is performed.

Local Variables: None

Subroutines Called: None

Calling Subroutines: PGM

Common Blocks: EDIT, TØP1, TØP2

Logic Flow: None

3.2.16 Subroutine: TREK (IT, KOUNT)

Purpose: To organize calls to the trajectory propagator and to evaluate target conditions.

Method: The trajectory propagator, TRAJ, performs two basic functions for TOPSEP: 1) trajectory integration from some specified starting time (TREF) to the stopping condition denoted by ISTØP, and 2) trajectory integration from the starting time to an event time (TEVNT). In the latter case TRAJ may be recalled and trajectory integration continued from the current event time to the next event time without requiring initialization of the trajectory routines and parameters. These capabilities are utilized in TOPSEP's submodes in different ways. For the simple trajectory propagation submode, TRAJ is required to integrate from the start time to the termination time. However, the targeting and grid submodes require that TRAJ return to TREK at certain phase times so that the s/c mass and state may be stored in blank common. This requirement is necessary only for the reference and trial trajectory when elements of THRUST(I, J) are used as controls. When TREK is called to set up grid trajectories and perturbed trajectories the appropriate mass and state are selected from blank common. TRAJ then integrates the trajectory from the beginning of the associated thrust phase to the terminal time thus avoiding the duplication of known trajectory segments. When elements of THRUST(I, J) are not used as controls, however, TRAJ integrates from the start time (TSTART) to the terminal time. TRAJ returns the s/c terminal state, and mass and the final time upon completion of the trajectory integration. To compute additional termination data or to compute target parameters such as BDT and BDR or orbital elements, subroutine BPLANE must be called. Subroutine TCØMP1 is then called to select and to store the appropriate target parameters in the vector TARPAR.

The flag returned from TRAJ which directs further computation of termination data is KUTØFF. The following table provides a summary of the KUTØFF options.

KUTØFF	Actual Stopping Condition	ISTØP	Requested Stopping Condition	Computed GØ TØ Statement Number
1	Final Time	1	Final Time	400
2	Final Time	2	Encounter	100
3	Final Time	3	SØI	100
4	Final Time	4	Stopping Radius	100
5	Encounter	2	Encounter	200
6	Encounter	3	SØI	200
7	SØI	3	SØI	300
8	Stopping Radius	4	Stopping Radius	400
9	Event Time	NA	Event Time	700

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
BIG	I	C	Constant equal to 1.E20
CA	I	C	Closest approach.
ECC	I	C	Eccentricity of orbit relative to the target planet at the actual stopping condition.
ICALL	O	C	Trajectory initialization flag.
IMØDE	I	C	TOPSEP submode designation.
INTEG	O	C	Flag indicating which equations are to be integrated in TRAJ.
IPRINT	O	C	Trajectory print flag.

Variable	Input/ Output	Argument/ Common	Definition
IT	I	A	Flag indicating type of initialization preceding the call to TRAJ.
ITP	O	C	Index of the target planet in the NB array (bodies included in the trajectory integration).
KMAX	I	C	Number of thrust controls (THRUST (I, J)) chosen to be elements of u.
KØUNT	I	A	Index on control.
KUTØFF	O	C	Termination flag.
LØCM	O	C	Blank common location of final S/C mass.
LØCRFM	I	C	Blank common location of the S/C masses evaluated at event times for the reference and all trial trajectories in a single iteration.
LØCTS	I	C	Blank common location of event times for the reference and all trial trajectories in a single iteration.
LØCXR	I	C	Blank common location of the 6-common state vectors associated with the event times of the reference and all the trial trajectories of a single iteration.
MEVENT	O	C	Flag designating trajectory propagation to event times.
MPRINT	I	C	Submode print option flags.
NPRI	O	C	Primary body designation.
NTNP	O	C	Vector of primary bodies associated with the event times of the reference and all trial trajectories in a single iteration.

Variable	Input/ Output	Argument/ Common	Definition
NTP	I	C	The target body code (NB (ITP)).
NTPH	I	C	Vector of control phase numbers associated with the event times of the reference and all trial trajectories in a single iteration.
NTPHAS	O	C	Thrust phase counter.
NTR	I	C	Trial trajectory counter.
NU	I	C	Number of controls.
RCA	O	C	Target planet encounter radius.
SCMASS	I/O	C	S/C mass at the trajectory start time.
SMA	O	C	Semi-major axis of the approach orbit relative to the target planet.
STATEO	I/O	C	S/C state at trajectory start time.
STØRE	I/O	C	Blank common variables.
TARPAR	O	C	Target values of the most recently generated trajectory.
TCA	O	C	Osculating time of closest approach.
TEVNT	O	C	Event time to be monitored by TRAJ.
TM	I	C	Number of seconds in a day.
TRCA	O	C	Time of closest approach determined by TRAJ if KUTØFF equals 5 or 6, otherwise set to TCA.
TREF	O	C	Reference time used by TRAJ to begin trajectory propagation.

Variable	Input/ Output	Argument/ Common	Definition
TSØI	O	C	Time at sphere of influence determined by TRAJ if KUTØFF equals 7, otherwise set to TSI.
TSTART	I	C	The reference trajectory start time.
TSTØP	O	C	The actual trajectory termination time.
UREL	O	C	Array containing the position components of the S/C relative to the bodies flagged in the NB array.
URELM	O	C	Vector containing the magnitude of the position components of the S/C relative to the bodies flagged in the NB array.
UTRUE	O	C	S/C position components relative to the primary body.
VCA	O	C	Osculating velocity at closest approach.
VRELM	O	C	Vector containing the magnitudes of the velocity components of the S/C relative to the bodies flagged in the NB array.
VTRUE	O	C	S/C velocity components relative to the primary body.
BDR	O	C	Osculating B-plane element orthogonal to the ecliptic plane.
BDR	O	C	Osculating B-plane element in the ecliptic plane.
IASTM	I	C	Flag designating the method of computing the target sensitivity matrix.

Variable	Input/ Output	Argument/ Common	Definition
LISTAR	I	C	Array of indices identifying selected target variables.
NT	I	C	Number of target variables.
TSI	O	C	Time of sphere of influence crossing based upon osculating B-plane conditions.
TUG	O	C	Logical flag determining whether injection conditions should be calculated.
VHP	O	C	Hyperbolic excess velocity.
VREL	I	C	Array containing the velocity components of the S/C relative to the bodies flagged in the NB array.

Local Variables

Variable	Definition
JUMP	Index on the thrust controls (THRUST (I, J)) chosen to be elements of $\underline{u}$.
MISS	Flag set to 1 if osculating elements are calculated outside the target planet's sphere of influence.
NPRIO	Primary body at time TSTART for the reference trajectory.
NTPHO	Thrust control phase number at time TSTART for the reference trajectory.
REFMO	S/C initial mass at time TSTART for the reference trajectory.
REFXO	S/C initial state at time TSTART for the reference trajectory.

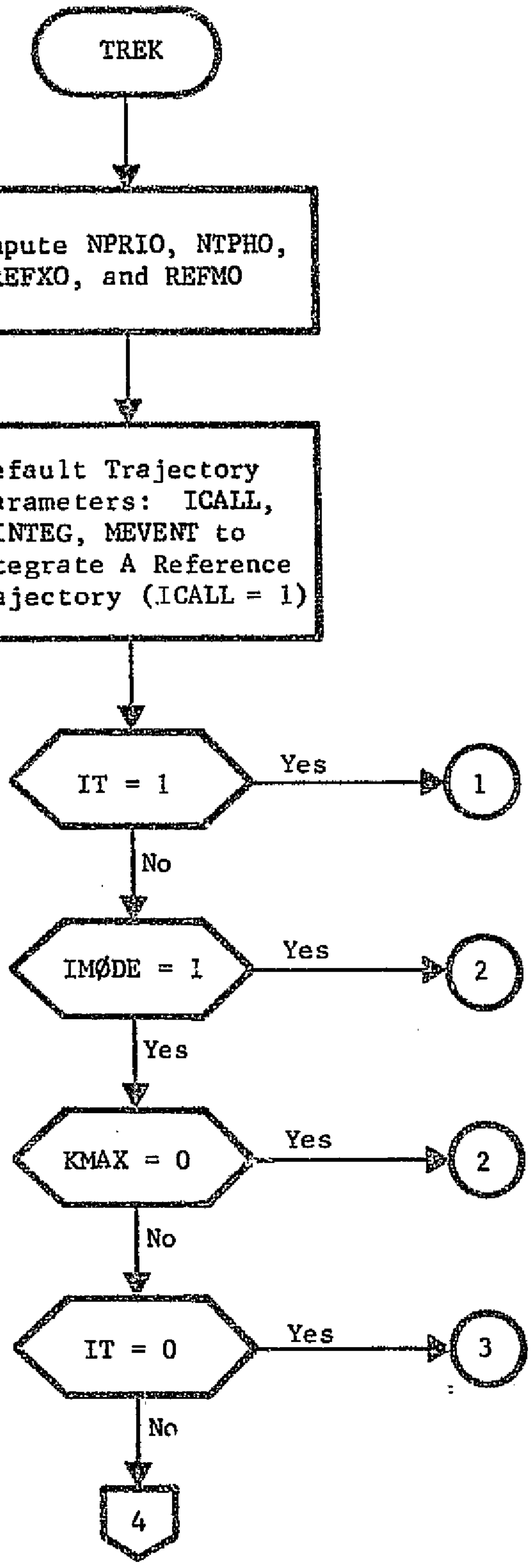
Subroutines Called: BPLANE, CØPY, VECMAG, TUGINJ, PRINT3, TCOMP1

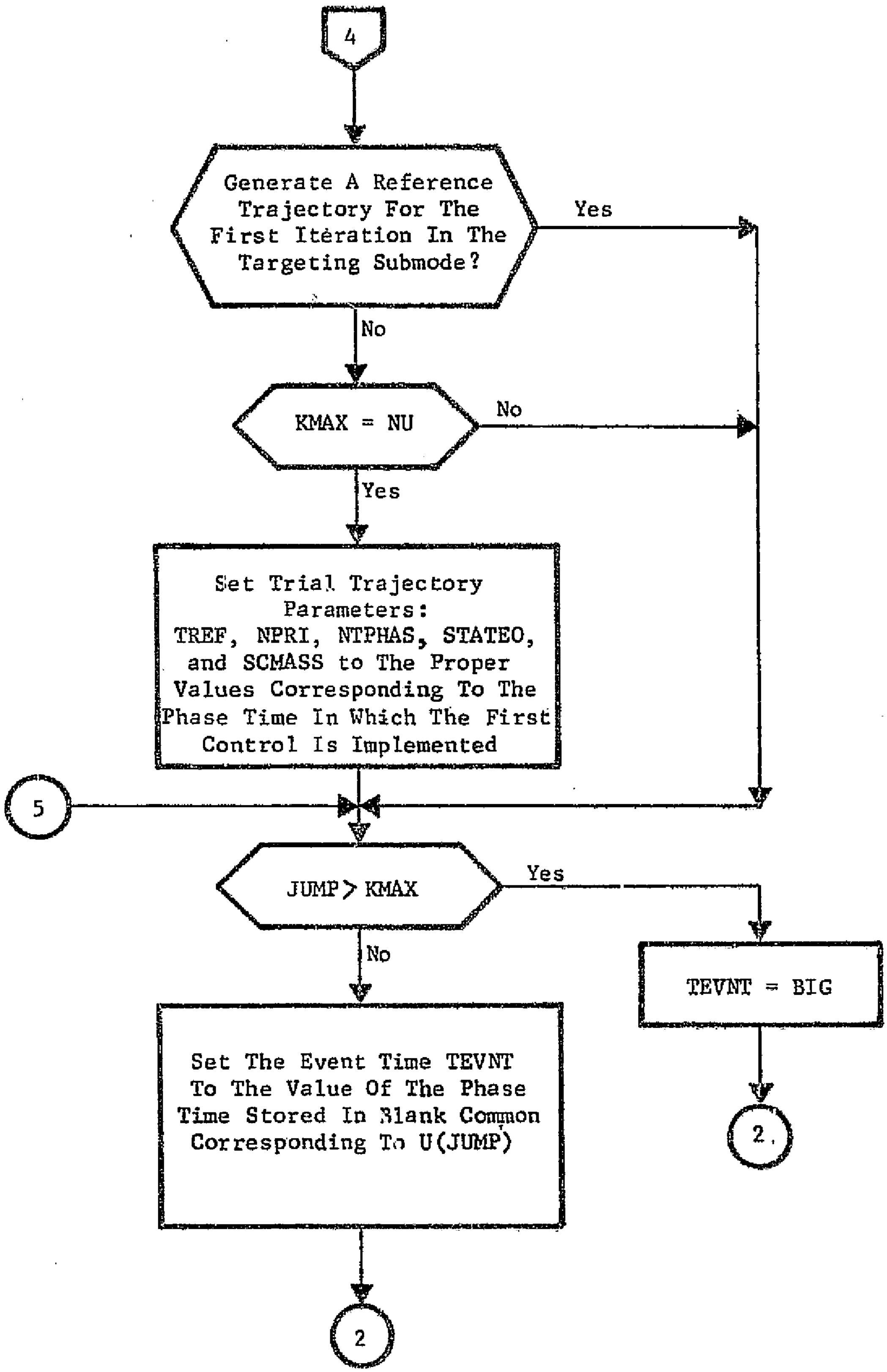
Calling Subroutines: FEGS, STMTAR

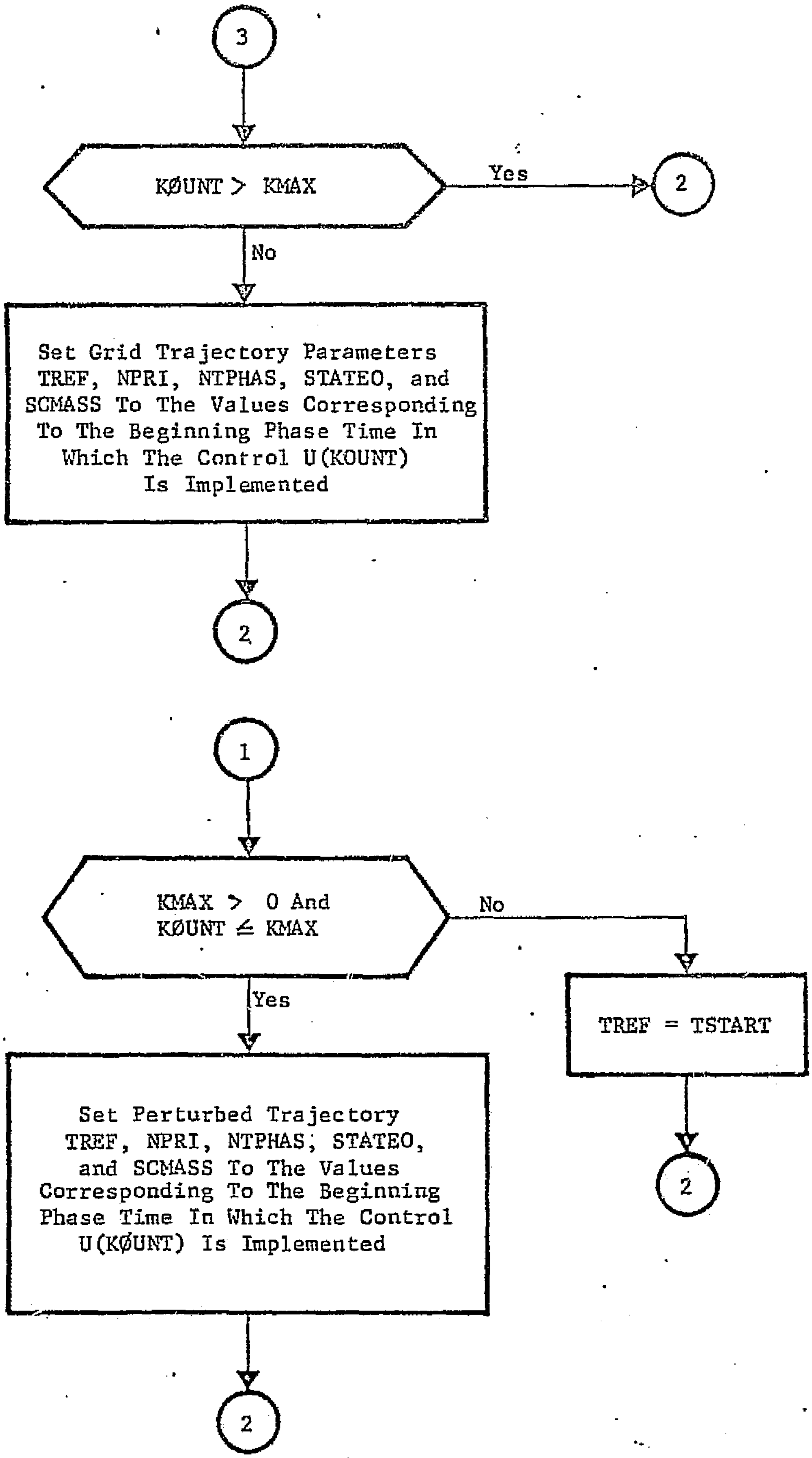
Common Blocks: (BLANK), CØNST, EDIT, EPHEM, GRID, PRINTH, TARGET, TIME, TØP1, TØP2, TRAJ1, TRAJ2, WORK, IASTM, TUG

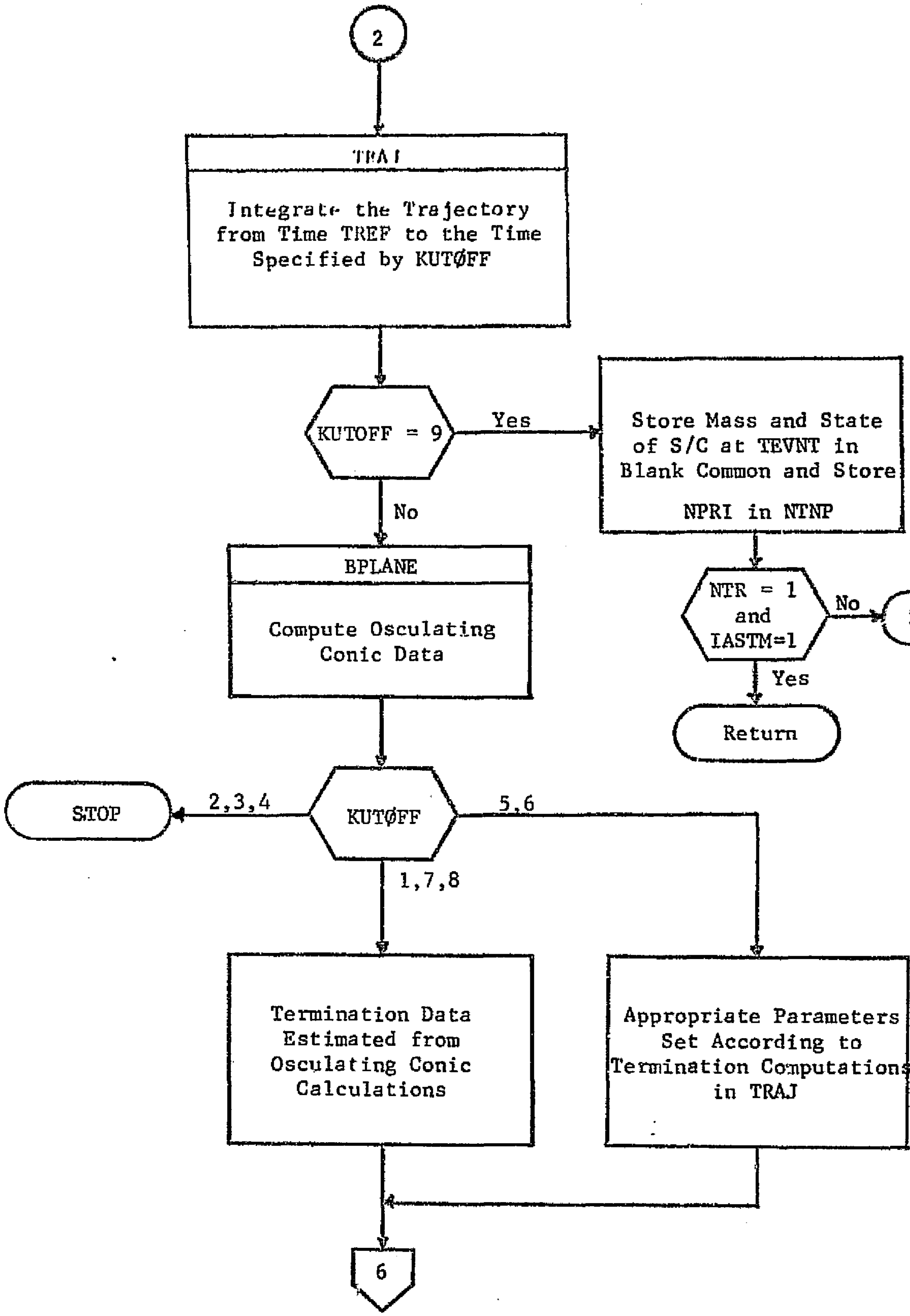
Page 182 has been deleted.

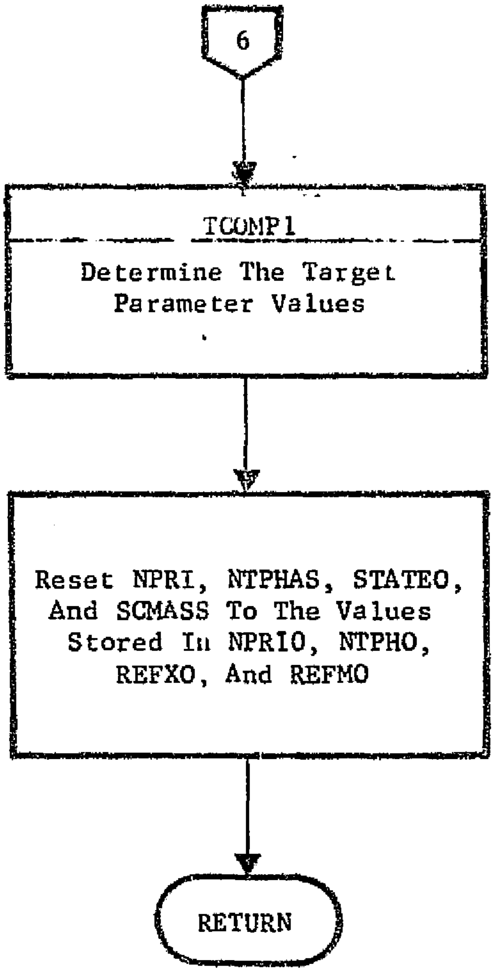
Logic Flow:


TREK
Compute NPRIO, NTPHO, REFXO, and REFMO
Default Trajectory Parameters: ICALL, INTEG, MEVENT to Integrate A Reference Trajectory (ICALL = 1)
IT = 1
Yes
1
No
IMØDE = 1
Yes
2
Yes
KMAX = 0
Yes
2
No
IT = 0
Yes
3
No
4


4
Generate A Reference Trajectory For The First Iteration In The Targeting Submode?
Yes
No
KMAX = NU
No
Yes
Set Trial Trajectory Parameters: TREF, NPRI, NTPHAS, STATEO, and SCMASS to The Proper Values Corresponding To The Phase Time In Which The First Control Is Implemented
5
JUMP > KMAX
Yes
No
TEVNT = BIG
2
Set The Event Time TEVNT To The Value Of The Phase Time Stored In Blank Common Corresponding To U(JUMP)
2


3
KØUNT > KMAX
Yes
2
No
Set Grid Trajectory Parameters TREF, NPRI, NTPHAS, STATEO, and SCMASS To The Values Corresponding To The Beginning Phase Time In Which The Control U(KOUNT) Is Implemented
2
1
KMAX > 0 And KØUNT ≤ KMAX
No
TREF = TSTART
2
Yes
Set Perturbed Trajectory TREF, NPRI, NTPHAS, STATEO, and SCMASS To The Values Corresponding To The Beginning Phase Time In Which The Control U(KØUNT) Is Implemented
2


2
TRAJ
Integrate the Trajectory from Time TREF to the Time Specified by KUTØFF
KUTOFF = 9
Yes
Store Mass and State of S/C at TEVNT in Blank Common and Store NPRI in NTNP
NTR = 1 and IASTM=1
No
5
Yes
Return
No
BPLANE
Compute Osculating Conic Data
STOP
2,3,4
KUTØFF
5,6
1,7,8
Termination Data Estimated from Osculating Conic Calculations
Appropriate Parameters Set According to Termination Computations in TRAJ
6


6
TCOMP1
Determine The Target Parameter Values
Reset NPRI, NTPHAS, STATEO, And SCMASS To The Values Stored In NPRIO, NTPHO, REFXO, And REFMO
RETURN

3.2.17 Subroutine: WEIGHT (DU1, DU2, DU, SINV, WG, WS, WU, NUD, NTD)

Entry Points: UNWATE

Purpose: To perform the appropriate control and target space transformations by weighting and unweighting the controls, gradients, sensitivities, and targets.

Method: Several different weighting algorithms have been devised to transform the control and target spaces in order to facilitate targeting and optimization. The weights are applied to "condition" the effects of selected controls when targeting and optimizing. The weighting algorithms are as follows:

1. User input weighting

$$WU\ (J) = \frac{1}{UWATE\,(J)}$$

2. Unitized control weighting

$$WU\ (J) = \left| \frac{1}{U(J) * UWATE\,(J)} \right|$$

3. Sensitivity weighting

$$WU\ (J) = MAX \left\{ \left| \frac{S(I,\ J)}{UWATE\,(J)} \right| \quad ,\ i = 1,\ NT \right\}$$

4. Combined sensitivity, target error, and control weighting

$$WU\ (J) = \sum_{I=1}^{NT} \left| \frac{S\ (I,\ J) * ETR\ (I,\ 1)}{U\ (J) * UWATE(J)} \right|$$

5. Target gradient weighting

$$G2\ (J) = 2 \sum_{I=1}^{NT} S\ (I,\ J) * ETR\ (I,\ 1)$$

$$WU\ (J) = \frac{|G2\ (J)|}{\sqrt{G2^T\ G2}}$$

6. Averaged gradient and control weighting

$$WU\ (J) = \frac{(10 * U(I) * UWATE(J) + \frac{.1}{G2(J)})}{(UWATE(J)\ * U(J)^2 + \frac{1}{G2(J)^2}\ )}$$

Remarks: This routine is used to weight controls and targets before the control correction is calculated and to unweight the same variables and certain additional parameters before the trial trajectories are made.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
DPSI	I	C	Target error to be removed during current iteration.
DU	I/O	A	Total control correction.

Variable	Input/ Output	Argument/ Common	Definition
DU1	I/O	A	Performance correction.
DU2	I/O	A	Constraint correction.
ETØL	I	C	Target tolerances.
ETR	I	C	Array of trial trajectory errors.
G	I	C	Performance gradient.
IWATE	I	C	Flag specifying type of weighting. 1, User input weighting 2, Unitized control weighting 3, Sensitivity weighting 4, Combined sensitivity, target error, and control weighting 5, Target gradient weighting 6, Averaged gradient and control weighting
IWATE	I	C	Flag specifying target weighting.
NT	I	C	Number of targets.
NTD	I	A	Integer variable used to dimension arrays in the argument list (number of targets).
NU	I	C	Number of controls.
NUD	I	A	Integer variable used to dimension arrays in the argument list.

Variable	Input/ Output	Argument/ Common	Definition
S	I	C	The sensitivity of targets to changes in controls.
SINV	I/O	A	Pseudo inverse of the sensitivity matrix.
U	I/O	C	The control vector.
UWATE	I	C	User input weights on controls (used in each weighting algorithm).
WG	Ø	A	Weighted performance gradient.
WØRK	I	C	Temporary working storage.
WS	O	A	Weighted sensitivity matrix.
WU	I	A	Control weighting vector.

Local Variables:

Variable	Definition
G2	Target gradient.
G2MAG	Magnitude of the target gradient.
STØRE	Temporary storage location.

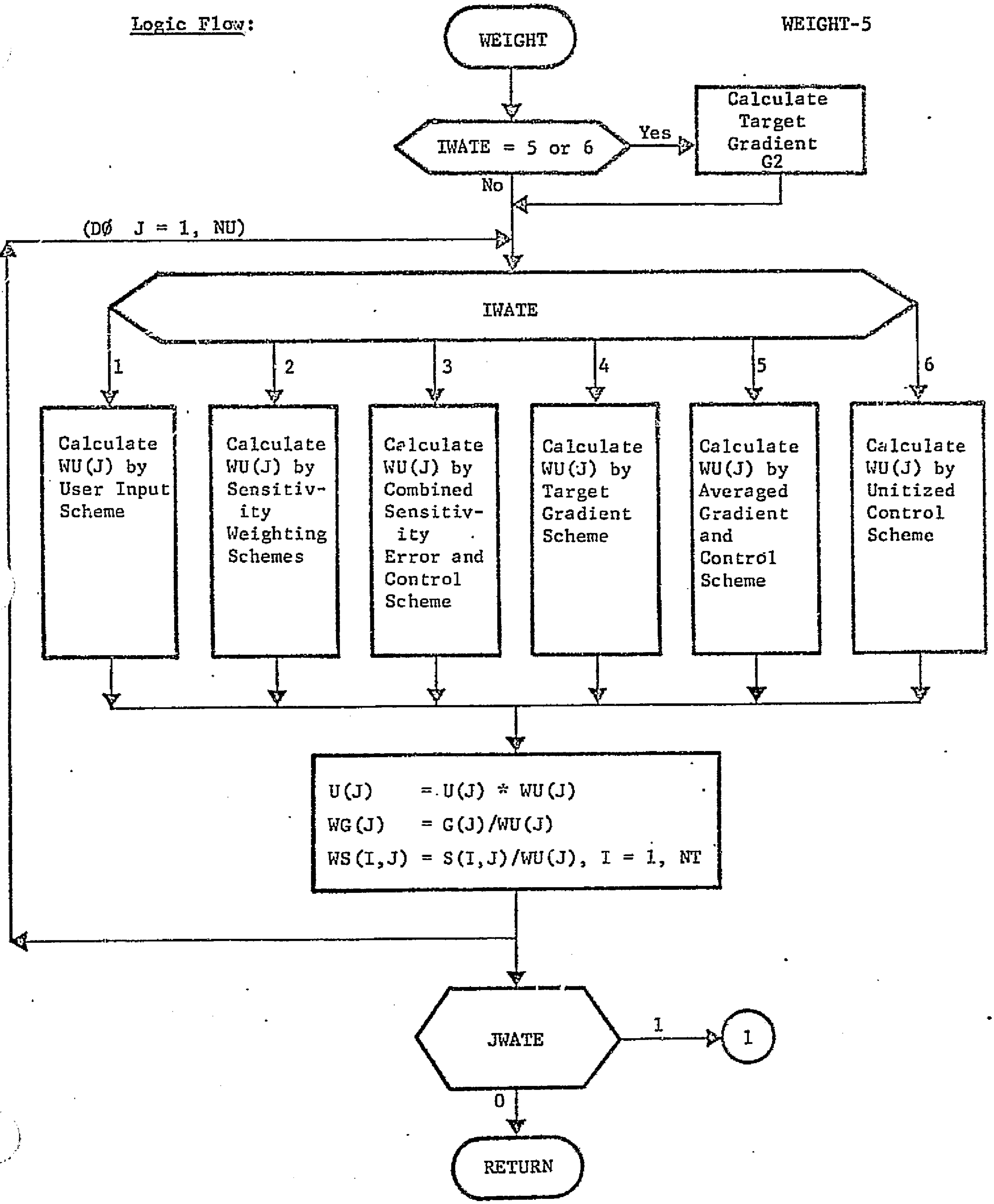
Subroutines Called: AMAX1, MMATB

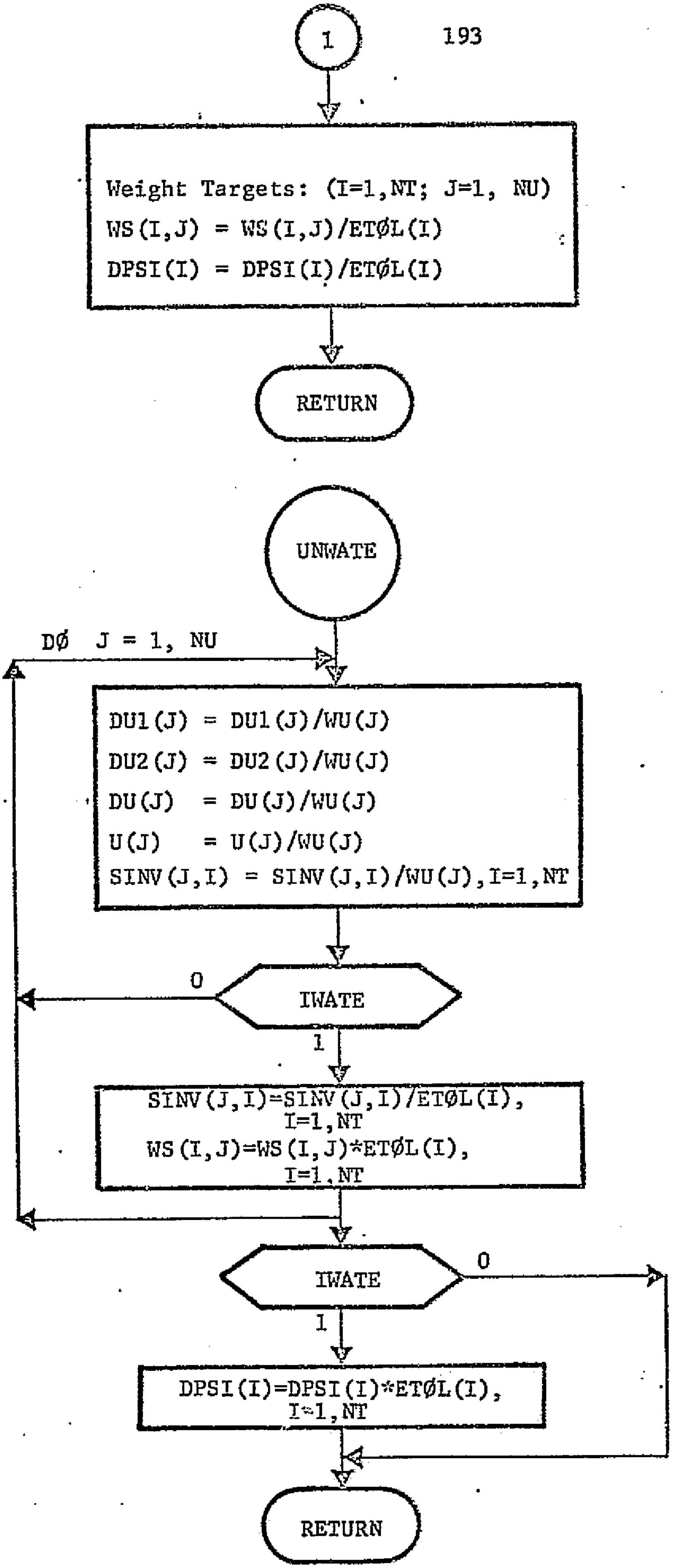
Calling Subroutines: SIZE

Common Blocks: EDIT, TØP1, TØP2, WØRK

Logic Flow:

Logic Flow: WEIGHT-5




1
Weight Targets: (I=1,NT; J=1, NU)
WS(I,J) = WS(I,J)/ETØL(I)
DPSI(I) = DPSI(I)/ETØL(I)
RETURN
UNWATE
DØ J = 1, NU
DU1(J) = DU1(J)/WU(J)
DU2(J) = DU2(J)/WU(J)
DU(J) = DU(J)/WU(J)
U(J) = U(J)/WU(J)
SINV(J,I) = SINV(J,I)/WU(J),I=1,NT
0
IWATE
1
SINV(J,I)=SINV(J,I)/ETØL(I), I=1,NT
WS(I,J)=WS(I,J)*ETØL(I), I=1,NT
IWATE
0
1
DPSI(I)=DPSI(I)*ETØL(I), I=1,NT
RETURN

3.3 Program: GØDSEP

Purpose: Executive control for error analysis.

Input/Output: Inputs are all trajectory data provided by DATAM. Outputs are all error analysis data.

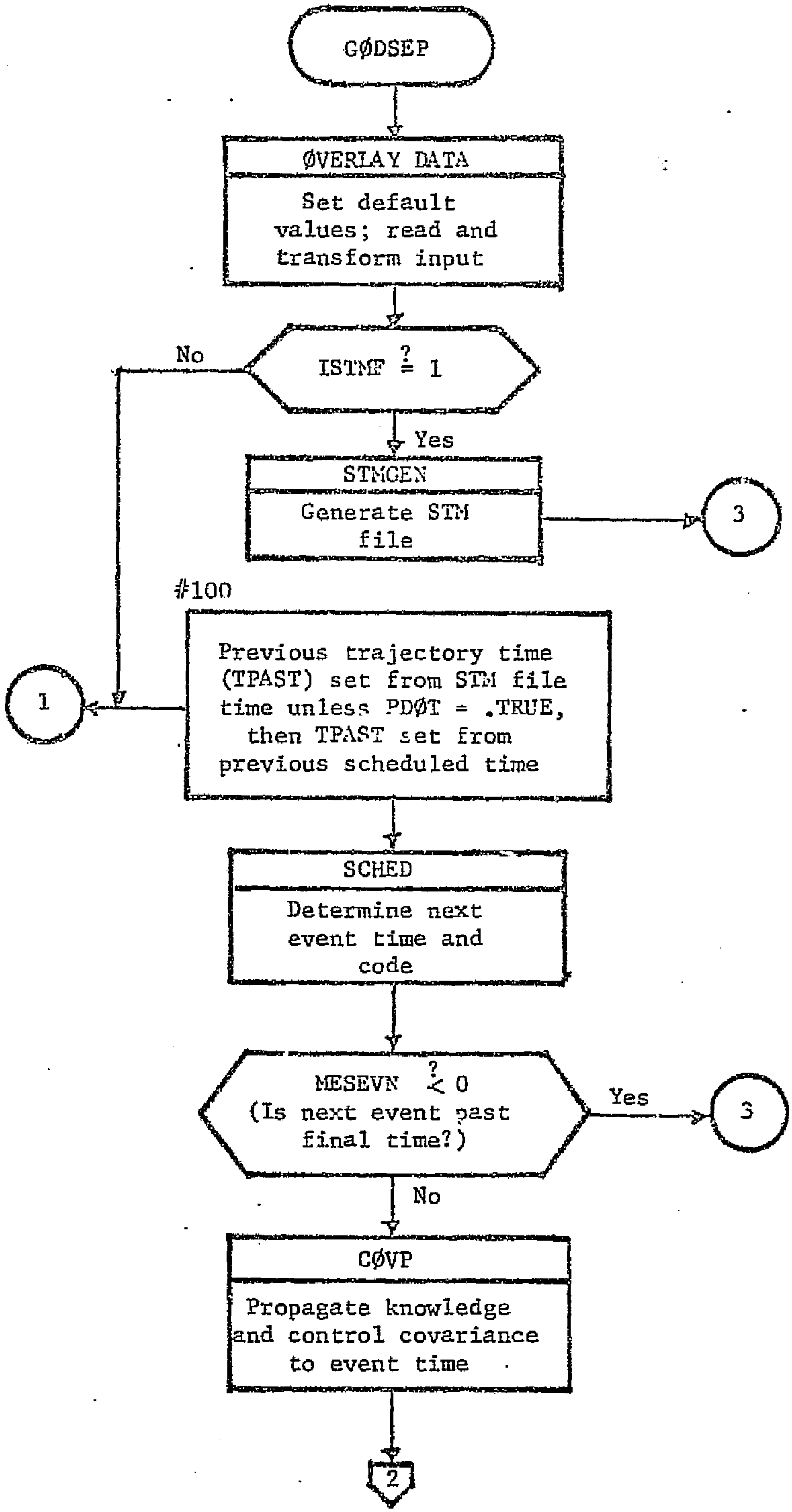
Local Variables: None

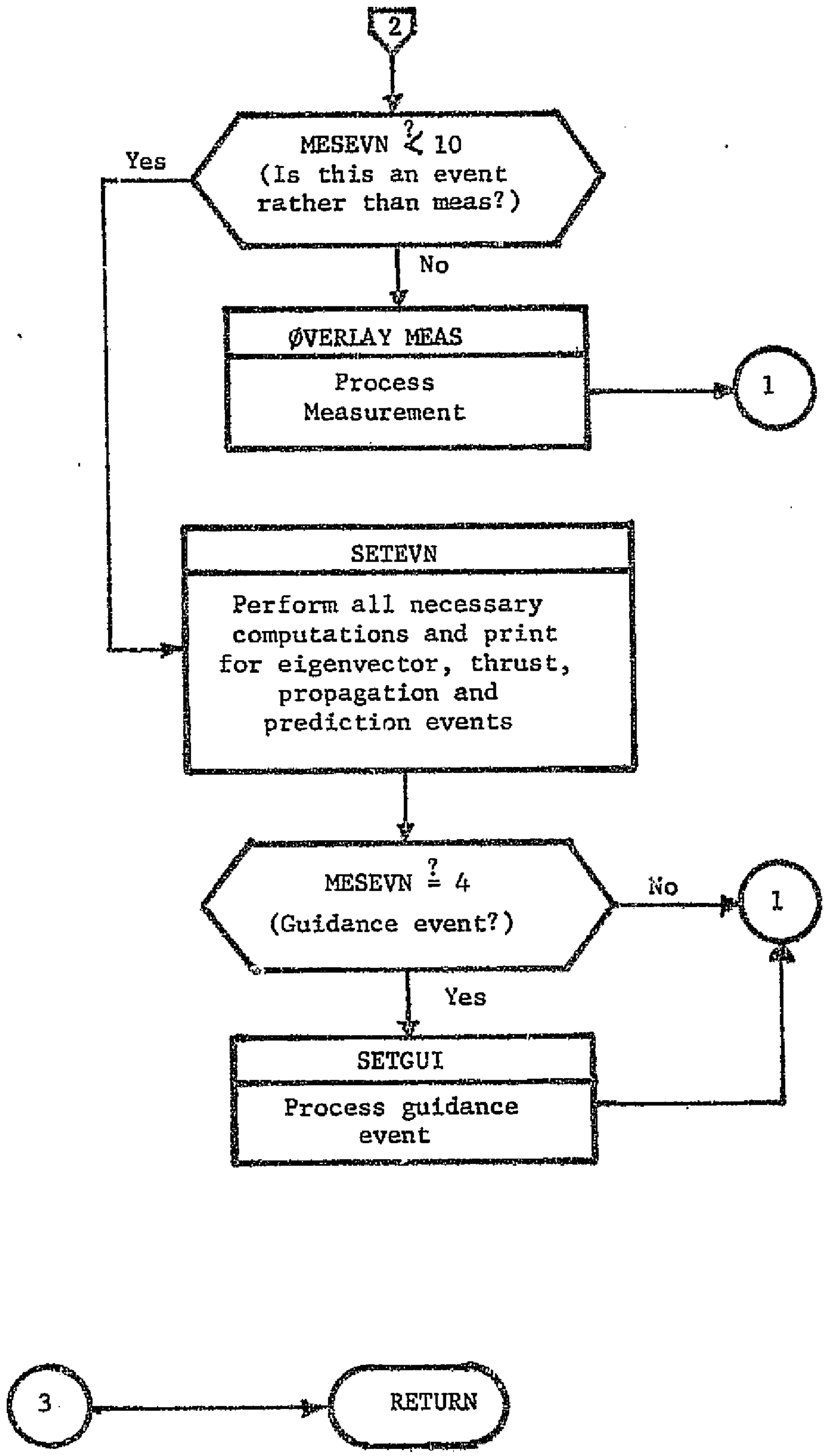
Subroutines Called: BLKDTG, CØPY, CØVP, DUMP, MASSIG, MATØUT, SCHED, SETEVN, SETGUI, STMGEN

Calling Subroutines: MAPSEP

Common Blocks: WORK, (BLANK), DIMENS, EDIT, ENCØN, LABEL, LØCATE, LØGIC, SCHEDI, SCHEDR, TRAJ1, TRAJ2

Logic Flow:


GØDSEP
ØVERLAY DATA
Set default values; read and transform input
ISTMF ?= 1
No
Yes
STMGEN
Generate STM file
3
#100
Previous trajectory time (TPAST) set from STM file time unless PDØT = .TRUE, then TPAST set from previous scheduled time
1
SCHED
Determine next event time and code
MESEVN ?< 0
(Is next event past final time?)
Yes
3
No
CØVP
Propagate knowledge and control covariance to event time
2


2
MESEVN ?< 10
(Is this an event
rather than meas?)
Yes
No
ØVERLAY MEAS
Process
Measurement
1
SETEVN
Perform all necessary
computations and print
for eigenvector, thrust,
propagation and
prediction events
MESEVN ?= 4
(Guidance event?)
No
1
Yes
SETGUI
Process guidance
event
3
RETURN

3.3.0 Subroutine: ASTØBS (HECP, HECE, TCURR, HMAT)

Purpose: To compute the observation partials for astronomical observations of the target body.

Method: See Volume I, Analytic Manual, Section 6.3.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
HECP	I	A	Heliocentric ecliptic coordinates of the ephemeris body.
HECE	I	A	Heliocentric ecliptic coordinates of the earth.
TCURR	I	A	Current trajectory time.
HMAT	O	A	Observation matrix.
GHZERØ	I	C	Greenwich hour angle evaluated at launch (TLNCH).
ØMEGAG	I	C	Rotational rate of the earth.
STALØC	I	C	Tracking Station/ Observatory locations.
ECEQ	I	C	Ecliptic to equational transformation matrix.

Local Variables:

Variables	Definition
GRILØN	Greenwich hour angle at TCURR.
GEQSTA	Geocentric equatorial coordinates of the observatory.
GECSTA	Geocentric ecliptic coordinates of the observatory.
RHØEC	Topocentric ecliptic position of the ephemeris body.

Local Variables:

Variables	Definition
RHØEQ	Topocentric equatorial position of the ephemeris body.
AEQ	Geocentric equatorial right ascension of the ephemeris body.
DEQ	Geocentric equatorial declination of the ephemeris body.
HH	Temporary storage for the observation matrix.

Subroutines Called: CYEQEC, LØCLST, LØDCØL, MMAT3, VECMAG, ZERØM

Calling Subroutines: ØBSERV

Common Blocks: CØNST, DIMENS, KEPCØN, MEASI, MEASR, TRKDAT, WØRK

Logic Flow: None

3.3.1 Subroutine: AUGCNV (CØVIN. CØVØUT, IØPT)

Purpose: To convert internal storage format of the augmented state covariance information from "block" (see Remarks) to augmented (see Remarks) form.

Remarks: The augmented covariance form is assumed as follows, where the individual matrix partitions or subblocks are defined in Input (Vol. II, User's Manual, Sec. 2.3):

$$\begin{bmatrix} P & CXS & CXU & CXV & CXW \\ CXS^T & PS & CSU & CSV & CSW \\ CXU^T & CSU^T & PU & CUV & CUW \\ CXV^T & CSV^T & CUV^T & PV & CVW \\ CXW^T & CSW^T & CUW^T & CVW^T & PW \end{bmatrix}$$

The "block" form assumes that all active partitions are stored contiguously in packed form in the following order:

P, CXS, CXU, CXV, CXW, PS, CSU, CSV, CSW, PU, CUV, CUW, PV, CVW, PW.

CØVIN and CØVØUT may share the same location. Therefore, in order to prevent writing over elements which have not been properly relocated in going from block to augmented form, PW is relocated first, then CVW and so on up the above-mentioned ordering of the block form. For the same reason, in going from augmented to block form the forward ordering (P, CXS, etc.) sequence is followed in relocating.

Input/Output:

Variables	Input/ Output	Argument/ Common	Definition
CØVIN	I	A	Augmented covariance in either block or augmented form according to IØPT
CØVØUT	O	A	Augmented covariance in opposite form from CØVIN, according to IØPT
IØPT	I	A	Conversion control flag =1, augmented to block form =-1, block to augmented form
LØCAUG	I	C	Array locating first word of each covariance partition within augmented form
LØCBLK	I	C	Array locating first word of each covariance partition within block form
NAUG	I	C	Length of augmented state vector
NDIM	I	C	Array of lengths of individual state vector partitions

Local Variables:

Variable	Definition
ISUB, JSUB	Subscripts used for locating elements at LØCAUG and LØCBLK
NCØL	Number of columns in current covariance sub-block
NRØW	Number of rows in current covariance sub-block

Subroutines Called: MPAK, MUNPAK, SYMUP

Calling Subroutines: PPAK

Common Blocks: WØRK, DIMENS

Logic Flow: None

3.3.2 Subroutine: BLKDTG

Purpose: To initialize label arrays in common /LABEL/ by DATA statements.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
AUGLAB	0	C	Augmented state vector element labels
EVLAB	0	C	Event labels
MESLAB	0	C	Measurement labels
PGLAB	0	C	Control covariance partition labels
PLAB	0	C	Knowledge covariance partirion labels
VECLAB	0	C	Augmented state vector partition labels

Local Variables: None

Subroutines Called: None

Calling Subroutines: GØDSEP

Common Blocks: LABEL

Logic Flow: None

3.3.3 Subroutine: BØMB

Purpose: To force abnormal termination with traceback.

Method: BØMB computes and attempts to use the square root of -1.0.

Remarks: On CDC 6000 series computers any attempt to use the square root of a negative number when operating with real variables causes program termination and provides a traceback to the main program of subroutines called and the location called from each. BØMB is called from several places in GØDSEP and its associated secondary overlays to indicate an unresolvable conflict of control variables.

Input/Output: None

Local Variables: None

Subroutines Called: None

Calling Subroutines: STMRDR, GAINF, DEFALT, DIMENS, NMLIST, ØUTPTG

Common Blocks: None

Logic Flow: None

3.3.4 Subroutine: CØRREL (PVAR, IØPTN, PUNCH, CØVLAB)

Purpose: To compute, print, and optionally, punch standard deviations and correlations coefficients from an input covariance matrix.

Remarks: Since VARSD (covariance to standard deviations and correlation coefficients) operates strictly on the upper triangle of a covariance matrix, only the diagonal of PVAR need be saved outside PVAR. The remaining lower triangle terms are then copied into the upper triangle.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
PVAR	I	A	Input covariance matrix.
IØPTN	I	A	Option flag. = 1, PVAR in covariance form = -1, PVAR already in standard deviations and correlation coefficients
PUNCH	I	A	Logical flag indicating if standard deviations and correlation coefficients are to be punched.
CØVLAB	I	A	Array of labels to be used for punching, if PUNCH = .TRUE.
AUGLAB	I	C	Augmented state vector labels.

Variable	Input/ Output	Argument/ Common	Definition
LØCAUG	I	C	Array locating partitions of augmented covariance matrix.
LØCLAB	I	C	Array locating state vector partition labels in AUGLAB.
NAUG	I	C	Length of augmented state vector.
NDIM	I	C	Array of dimensions of augmented state vector partitions.
PRNCØV	I	C	Logical array denoting which partitions of standard deviations and correlation coefficients are to be printed.

Local Variables:

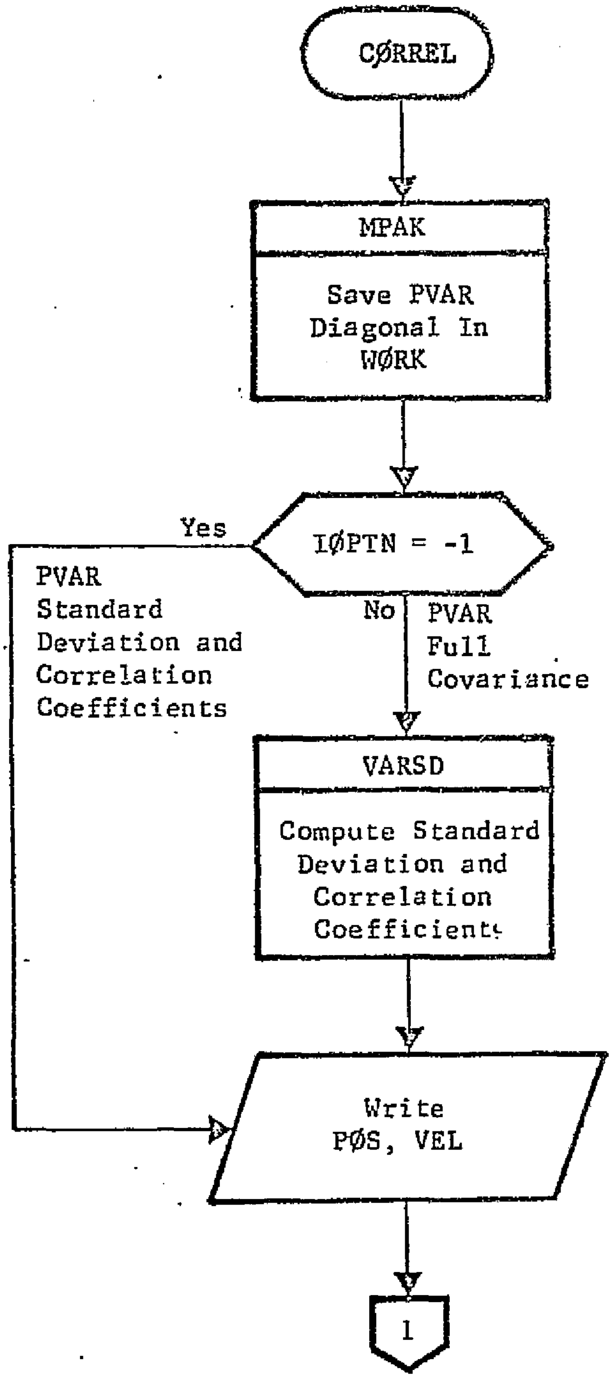
Variable	Definition
PØS	1 σ RSS position uncertainty.
VEL	1 σ RSS velocity uncertainty.

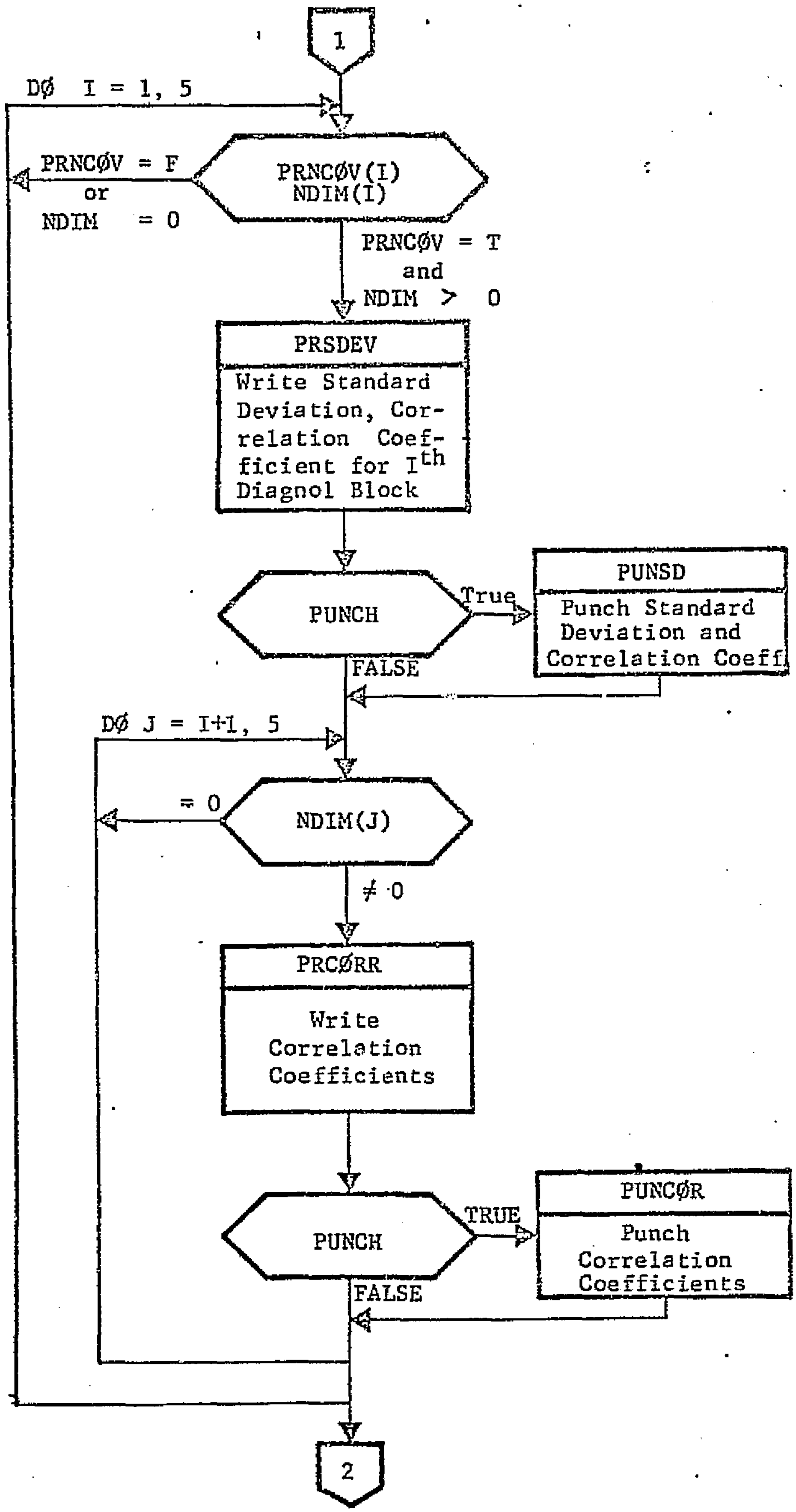
Subroutines Called: MPAK, VECMAG, VARSD, PRSDEV, PUNSD, PRCØRR, PUNCØR, SYMLØ, MUNPAK

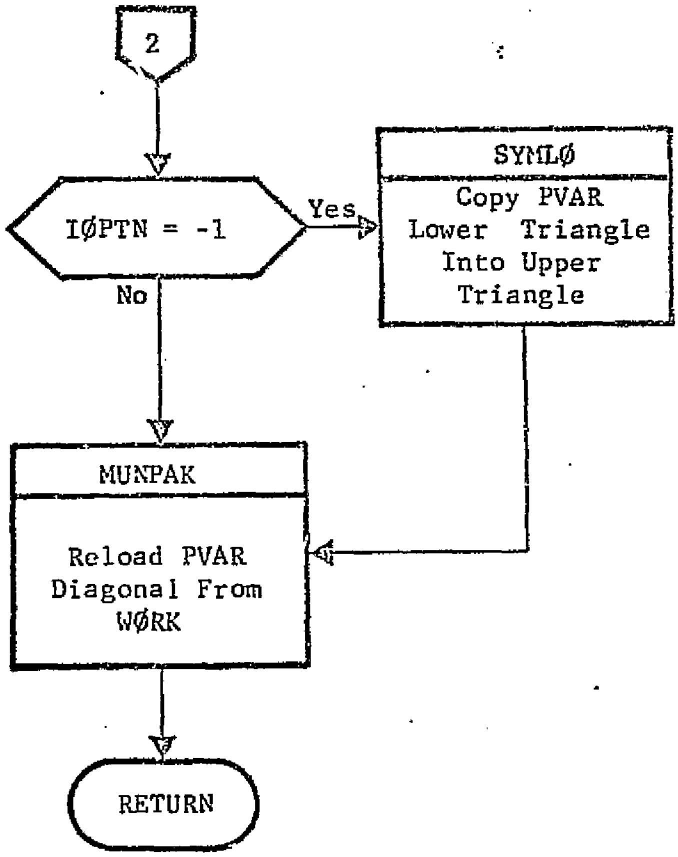
Calling Subroutines: SETEVN, GUIDE, MEASPR

Common Blocks: WØRK, DIMENS, LABEL, LØGIC

Logic Flow:


CØRREL
MPAK
Save PVAR Diagonal In WØRK
IØPTN = -1
Yes
PVAR Standard Deviation and Correlation Coefficients
No
PVAR Full Covariance
VARSD
Compute Standard Deviation and Correlation Coefficients
Write PØS, VEL
1


1
DØ I = 1, 5
PRNCØV = F
or
NDIM = 0
PRNCØV(I)
NDIM(I)
PRNCØV = T
and
NDIM > 0
PRSDEV
Write Standard Deviation, Correlation Coefficient for I th Diagnol Block
PUNCH
True
PUNSD
Punch Standard Deviation and Correlation Coeff
FALSE
DØ J = I+1, 5
= 0
NDIM(J)
≠ 0
PRCØRR
Write Correlation Coefficients
PUNCH
TRUE
PUNCØR
Punch Correlation Coefficients
FALSE
2


2
IØPTN = -1
Yes
No
SYMLØ
Copy PVAR Lower Triangle Into Upper Triangle
MUNPAK
Reload PVAR Diagonal From WØRK
RETURN

3.3.5 Subroutine: CØVP (T, TF, STMRD, PIN)

Purpose: To propagate a covariance between two time points.

Method: Three options are available:

1) propagation by transition matrices read from STM file;
2) propagation by transition matrices computed as needed and not saved; or
3) propagation by integration of covariance variational equations.

Remarks: Independent of propagation method, the output of CØVP is always stored in blank common located by the integer variable PTEMP. This is true even for zero length propagation intervals, in which case the input covariance is merely copied to that location.

Additionally, when the option to read the STM file is exercised, CØVP automatically propagates the control covariance if control propagation is indicated (logical variable PRØPG).

When CØVP is called with both STMRD and PDØT false (nominally for prediction events only) tests are made to subdivide the complete propagation interval into as many subintervals as necessary to guarantee that no transition matrix propagation crosses a thrust phase change, since that would violate effective process noise model assumptions.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
T	I	A	Beginning time of propagation interval
TF	I	A	End time of propagation interval
STMRD	I	A	Logical variable indicating source of transition matrices if transition matrices are to be used =T, read transition matrices from STM file =F, generate transition matrices by calling TRAJ overlay
PIN	I	A	Input augmented covariance
DELTIM	I/O	C	Propagation interval length
DXDKST	O	C	Keplerian to cartesian transformation for ephemeris body

Variable	Input/ Output	Argument/ Common	Definition
DYNØIS	I	C	Dynamic noise flag
GT	I/O	C	Transformation matrix from thrust cone-clock system to heliocentric ecliptic coordinates evaluated at end of propagation interval
GTSAVE	O	C	Same transformation matrix as GT, but evaluated at beginning of propagation interval
IAUGDC	I	C	Dynamic augmentation vector
ICALL	O	C	Initialization parameter for TRAJ (sec. 3.5)
IEP	I	C	Locator in UP, VP of elements corresponding to ephemeris planet
IEPHEM	I	C	Flag indicating form of ephemeris elements, if any
INTEG	O	C	Control parameter for TRAJ (sec. 3.5)
ISTØP	O	C	Control parameter for TRAJ (sec. 3.5)
LIST	I	C	Array of state vector augmentation parameter numbers

Variable	Input/ Output	Argument/ Common	Definition
LISTDY	I	C	Array of dynamic parameter numbers included in transition matrices
LØCFØ	I	C	Location in blank common of covariance matrix to be integrated when PDØT option is selected
LØCTC	I	C	Location in blank common of either transition matrix or covariance matrix returned by TRAJ (sec .5) after integration
LPDØT	I	C	Ordered list of parameters expected by TRAJ (sec. 3.5) when covariance integration option is selected. LPDØT is equivalenced to IGPØL array in common /SCHEDI/ since no guidance events are permitted when integrating covariance variational equations
MEVENT	O	C	Control flag for TRAJ (sec 3.5)

Variable	Input/ Output	Argument/ Common	Definition
NAUG	I	C	Length of total augmented state vector
NEPHEL	I	C	Number of ephemeris elements augmented to state vector
NTPHAS	I	C	Number of current thrust phase
PDØT	I	C	Logical flag =T, integrate covariance variational equations =F, propagate covariances by transition matrices
PG	I	C	Location in blank common of control covariance
PHI	I	C	Location in blank common of transition matrix
PLØCAL	I	C	Location in blank common of working storage block as large as the augmented covariance matrix
PRØPG	I	C	Logical flag, operative only if PDØT = FALSE and STMRD = TRUE =T, propagate control covariance simultaneously with knowledge

Variable	Input/ Output	Argument/ Common	Definition
			=F, do not propagate control covariance
Q	O	C	Effective process noise matrix
SMASS	I	C	Mass of Sun.
STATEO	O	C	Initial heliocentric ecliptic S/C state for TRAJ (sec 3.5) when ICALL = 1
TCURR	I	C	Current trajectory time
TEVNT	O	C	Event time for propagation (either of covariance or transition matrix) to by TRAJ (sec 3.5)
TG	I	C	Epoch of input control covariance referenced to TLNCH
TM	I	C	Conversion factor, seconds/day
TREF	O	C	Reference time for TRAJ (sec 3.5)
TTHRST	I	C	Array of thrust event times
UP	I	C	Array of n-body heliocentric position vectors

Variable	Input/ Output	Argument/ Common	Definition
UTRUE	I	C	S/C heliocentric position vector
VP	I	C	Array of n-body heliocentric velocity vectors
VTRUE	I	C	S/C heliocentric velocity vector

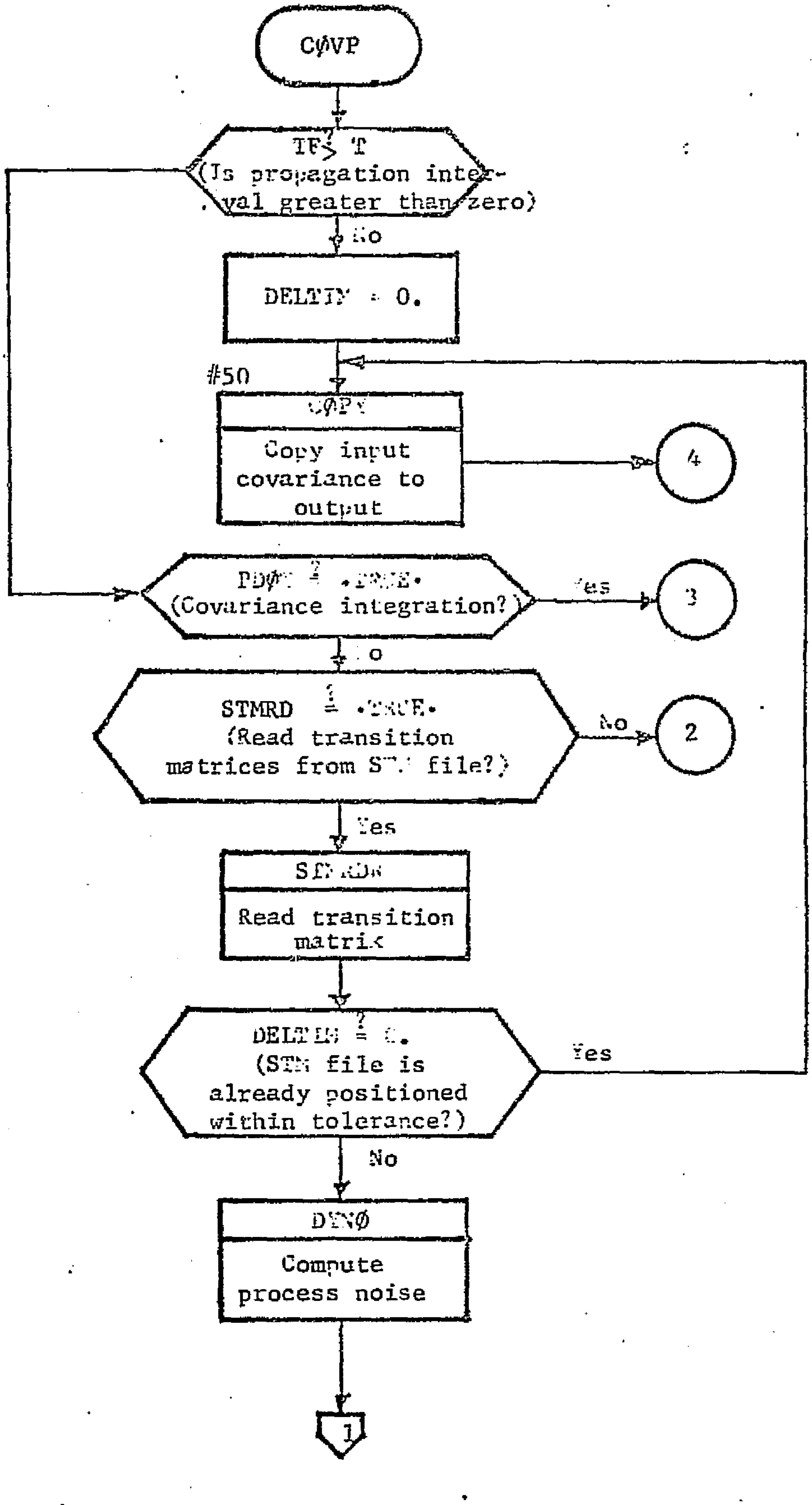
Local Variables:

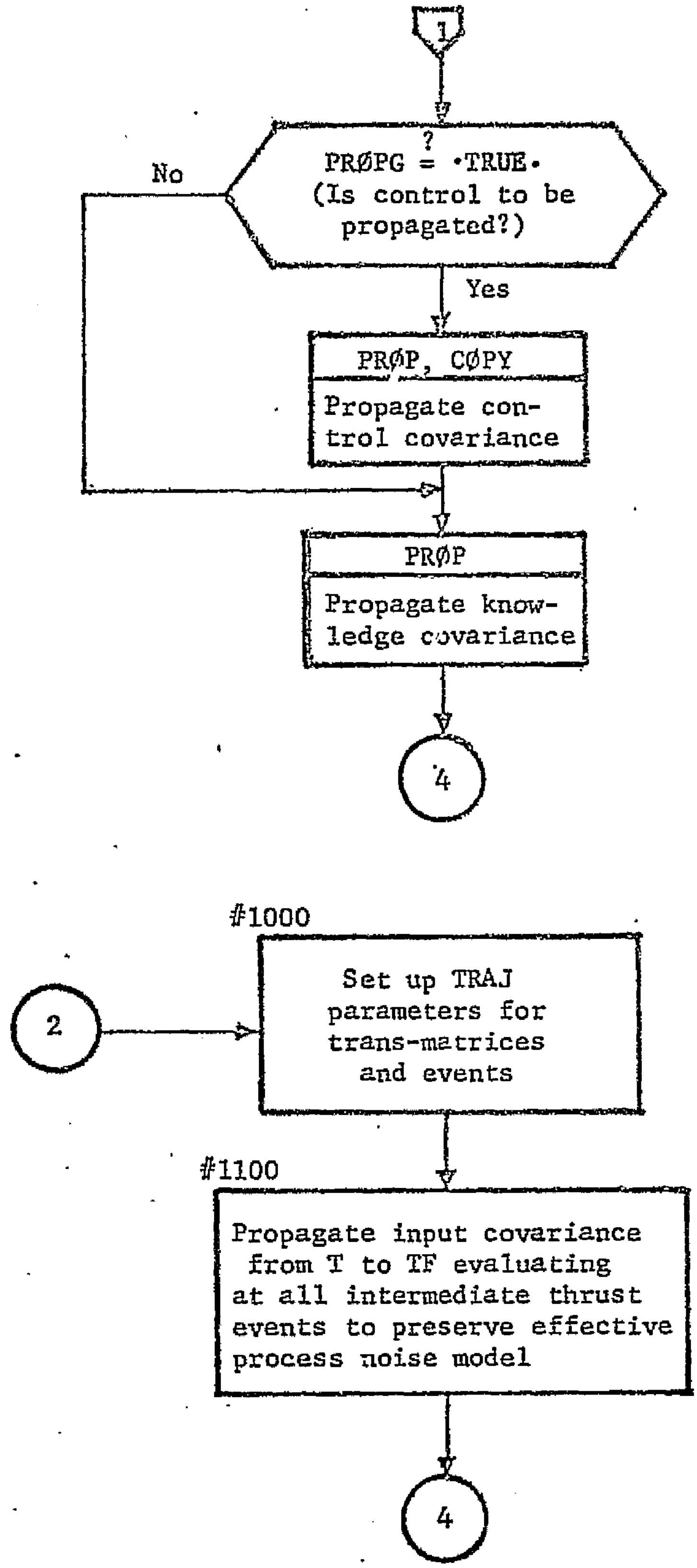
Variable	Definition
FRSTIM	Logical flag used when PDØT = TRUE to control one-time only initialization of parameters for TRAJ (sec 3.5) =T, first pass through CØVP =F, not first pass through CØVP
ILIST	List of augmented dynamic parameters
T1, T2	Start and stop times respectively for propagation subintervals as governed by thrust events (see Remarks)

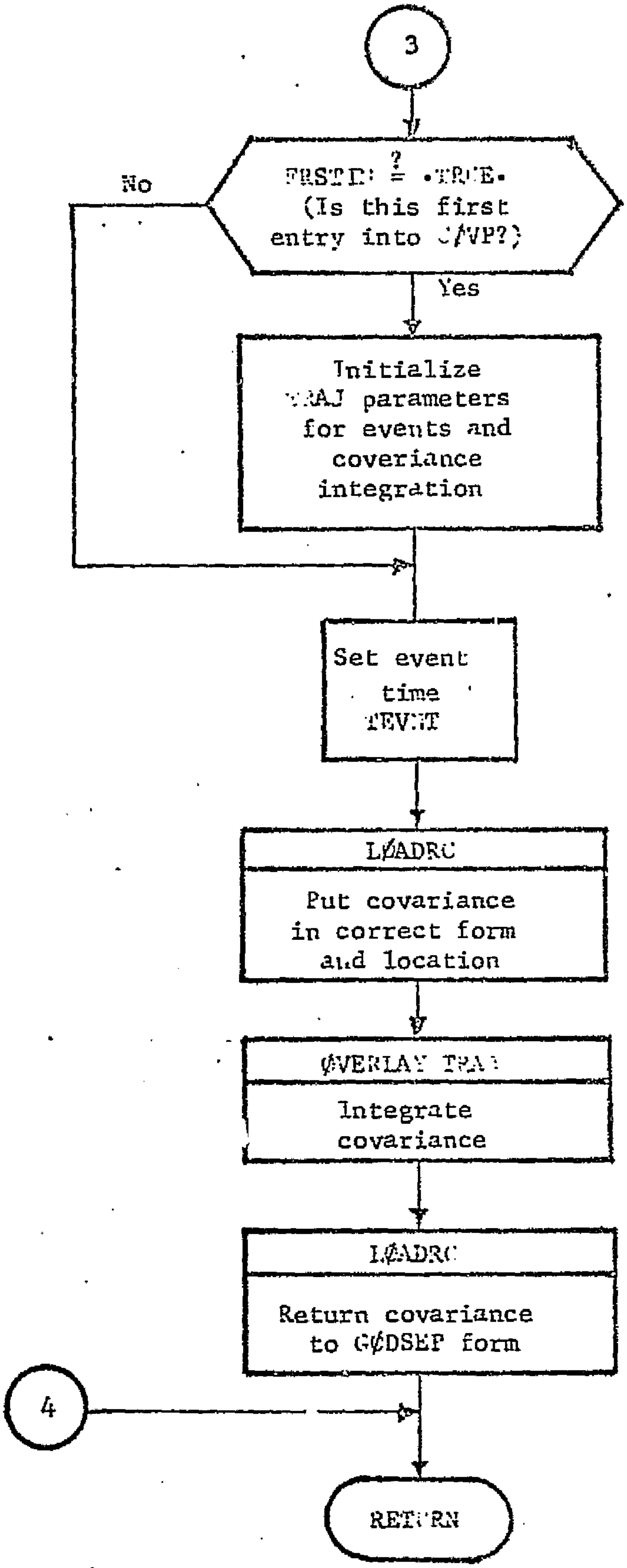
Subroutines Called: AMABT, CARKEP, CØPY, DYNØ, LØADRC, MMAB, MMABT, MUNPAK, PRØP, STMPR, STMRDR, STMUSE, ZERØM

Common Blocks: WØRK, (BLANK), CØNST, DIMENS, KEPCØN, LØCATE, LØGIC, MEASI, PRØPI, PRØPR, SCHEDI, SCHEDR, EPHEM, TIME, TRAJ1, TRAJ2

Logic Flow:


CØVP
TF ? T (Is propagation interval greater than zero)
No
DELTIM = 0.
#50
CØPY
Copy input covariance to output
4
PDØT ? .TRUE. (Covariance integration?)
Yes
3
No
STMRD ? .TRUE. (Read transition matrices from STM file?)
No
2
Yes
Read transition matrix
DELTIM ? 0. (STM file is already positioned within tolerance?)
Yes
No
DYNØ
Compute process noise
1


1
PRØPG ? = ·TRUE·
(Is control to be propagated?)
No
Yes
PRØP, CØPY
Propagate control covariance
PRØP
Propagate knowledge covariance
4
#1000
2
Set up TRAJ parameters for trans-matrices and events
#1100
Propagate input covariance from T to TF evaluating at all intermediate thrust events to preserve effective process noise model
4


3
FRSTEN ? = .TRUE.
(Is this first entry into CØVP?)
No
Yes
Initialize TRAJ parameters for events and coveriance integration
Set event time TEVNT
LØADRC
Put covariance in correct form and location
ØVERLAY TRAJ
Integrate covariance
LØADRC
Return covariance to GØDSEP form
4
RETURN

3.3.6 <u>Subroutine:</u> CYEQEC (STACYL, GRLØN, ECEQ, ØMEGA, GEQSTA, GECSTA)

<u>Purpose:</u> To compute instantaneous geocentric equatorial and geocentric ecliptic cartesian coordinates of a point (station location) given in geocentric equatorial cylindrical coordinates.

<u>Method:</u> Given cylindrical coordinates r_s, λ, z (spin radius, longitude, z - height) instantaneous Greenwich hour angle (G), and sidereal rotation rate, ω, equatorial coordinates are

$$x_{eq} = r_s \cos(\lambda + G)$$

$$y_{eq} = r_s \sin(\lambda + G)$$

$$z_{eq} = z$$

$$\dot{x}_{eq} = -\omega r_s \sin(\lambda + G) = -\omega y_{eq}$$

$$\dot{y}_{eq} = \omega r_s \cos(\lambda + G) = \omega x_{eq}$$

$$\dot{z}_{eq} = 0$$

Ecliptic position and velocity are computed by the application of the equatorial to ecliptic transformation $\partial x_{ec}/\partial x_{eq}$

$$\begin{bmatrix} x_{ec} \\ y_{ec} \\ z_{ec} \end{bmatrix} = \partial x_{ec}/\partial x_{eq} \cdot \begin{bmatrix} x_{eq} \\ y_{eq} \\ z_{eq} \end{bmatrix}$$

$$\begin{bmatrix} \dot{x}_{ec} \\ \dot{y}_{ec} \\ \dot{z}_{ec} \end{bmatrix} = \partial x_{ec}/\partial x_{eq} \begin{bmatrix} \dot{x}_{eq} \\ \dot{y}_{eq} \\ \dot{z}_{eq} \end{bmatrix}$$

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
STACYL	I	A	Station cylindrical co-ordinate (r_s, λ, z)
GRLØN	I	A	Instantaneous Greenwich hour angle
ECEQ	I	A	Equatorial to ecliptic transformation ($\partial x_{ec}/\partial x_{eq}$)
ØMEGA	I	A	Earth's sidereal rotation rate
GEQSTA	O	A	Station geocentric equatorial position and velocity
GECSTA	O	A	Station geocentric ecliptic position and velocity

Subroutines Called: None

Calling Subroutines: ØBSERV

Common Blocks: None

Logic Flow: None

3.3.7 Program: DATAG

Purpose: Executive control of GØDSEP data overlay.

Remarks: DATAG performs no computations. It merely calls three separate subroutines to break the data overlay coding into more easily managed blocks.

Input/Output: All initialization parameters for GØDSEP.

Local Variables: None

Subroutines Called: DEFALT, INPUTG, ØUTPTG

Calling Subroutines: GØDSEP

Common Blocks: None

Logic Flow: None

3.3.8 Subroutine: DEFALT

Purpose: To establish default values for all error analysis inputs.

Remarks: Only those variables not having default values defined in GØDSEP input (Vol. II, User's Manual, Section 2.3) will be included in the following Input/Output list.

Input/Output:

Variables	Input/ Output	Argument Common	Definition
EPØCH	I	C	Julian date of launch epoch
GHZERØ	O	C	Greenwich hour angle evaluated at time EPØCH
IAUGDC	I	C	Array of flags controlling dynamic parameter augmentation for transition matrices
IAUGPH	O	C	Location of ephemeris element flags in IAUG array
IAUGST	O	C	Location of station location parameter flags in IAUG array
IBAZEL	O	C	Location of azimuth and elevation angle measurement bias flags in IAUG array

Variables	Input/ Output	Argument Common	Definition
IBDIAM	O	C	Location of apparent planet diameter measurement bias flag in IAUG array
IBSTAR	O	C	Location of star-planet angle measurement bias flags in IAUG array
IB2WAY	O	C	Location of 2-way range and range-rate measurement bias flags in IAUG array
IB3WAY	O	C	Location of 3-way range and range-rate measurement bias flags in IAUG array
IDMAX	O	C	Maximum allowable parameter number for any dynamic parameter in IAUG array
IEP	O	C	Parameter used to locate ephemeris body position and velocity in UP, VP, UREL, VREL arrays (common /TRAJ2/)
ITP	O	C	Same as IEP, only for target body

Variables	Input/ Output	Argument Common	Definition
LIST	O	C	Array listing parameter numbers of augmented state vector. For first six locations (for basic S/C state) LIST(I) = -I
LISTDY	O	C	List of parameter numbers of all dynamic parameters augmented to S/C state for transition matrices. Defining values determined by IAUGDC array.
LØCS	O	C	Parameters locating first word of blank common available to TRAJ (sec. 3.5) default value, = 1
MAXAUG	O	C	Maximum allowable length of augmented state vector. Determined by dimensions of LIST and AUGLAB arrays. Default value, = 30.

Variables	Input/ Output	Argument Common	Definition
MAXDIM	O	C	Array of maximum allowable dimensions on individual state vector partitions. Values set are governed by dimensions of input covariance matrices in subroutine NMLIST (sec 3.3.25). Default values are: (1) = 6, S/C state vector (2) = 11, solve-for parameters (3) = 13, dynamic consider parameters (4) = 15, measurement consider parameters (5) = 10, ignore parameters
MAXSTA	O	C	Largest station number allowed for augmenting 2-way or 3-way range or range-rate bias to the S/C state vector
NPHSTM	O	C	Length of augmented state vector of dynamic parameters used in transition matrices

Variables	Input/ Output	Argument Common	Definition
ØMEGAG	O	C	Earth sidereal rotation rate default value = 6.300388099 rad/day
RAD	I	C	Conversion factor, degrees/ radian.
TEND	I	C	Trajectory end time in days referenced to EPØCH as defined in $TRAJ name- list (Vol. II, User's Manual, sec. 2.1)
THRUST	I	C	Array defining thrust con- trol policies, phase end times and specific param- eter values (see common /TRAJ1/)
TM	I	C	Conversion constant, seconds/ day
TSTART	I	C	Trajectory start time in days referenced to EPØCH, as defined in $TRAJ namelist (Vol. II, User's Manual, Sec. 2.1)

Variables	Input/ Output	Argument Common	Definition
XLAB	O	C	Six-character Hollerith labels corresponding to input parameters as defined by IAUG array (see Vol. II, User's Manual, Sec. 2.3)

Local Variables:

Variable	Definition
MAXPAR	Maximum number of parameters available for augmentation. Governed by dimensions of IAUG and XLAB arrays. Current default value = 50.
TFRAC	Fraction of a day the initial Julian date, EPØCH, is away from midnight Greenwich Mean Time. Used in computing GHZERØ.

Subroutine Called: BØMB, LØCATE

Calling Subroutines: DATAG

Common Blocks: WØRK, (BLANK), CØNST, DATAGI, DATAGR, DIMENS, GUIDE, KEPCØN, LABEL, LØCATE, LØGIC, MEASI, MEASR, PRØPI, PRØPR, SCHEDI, SCHEDR, TIME, TRKDAT, TRAJ1, TRAJ2

Logic Flow: None

3.3.9 Subroutine: DIMENS

Purpose: To define dimensions and locations of all matrices located in blank common.

Remarks: Blank common locations set aside by the variables PHI, PLØCAL and PTEMP are normally allocated the same number of words of storage as for a covariance matrix. There are, however, two exceptions to this standard. If the dimensions of transition matrices to be read from the STM file are greater than those of the augmented covariance matrix, or if both the transition matrices from the STM file and the augmented covariance are smaller than 9x9 and guidance events are to be executed. The second case requires a minimum 9x9 area since thrust bias sensitivities are required for low thrust guidance maneuver evaluations.

Since only one secondary overlay may reside in core at any one time, all blank common locations associated only with secondary overlays begin at the same address. Therefore, LØCS (trajectory), H (measurement) and PG1 (guidance) are set to the same location.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
AUGLAB	O	C	Hollerith labels for all parameters augmented to state vector.
CØNRD	I	C	Logical flag indicating if control uncertainties read in.
H	O	C	Location in blank common of observation matrix.
IAUG	I	C	Array of parameter augmentation flags.
IAUGDC	O	C	Dynamic parameter augmentation flags.
IAUGPH	I	C	Location in IAUG array of ephemeris element flags.
IDMAX	I	C	Maximum parameter number allowed for a dynamic parameter in IAUG array.
IGAIN	I	C	Integer flag for OD algorithm.
IGFØRM	I	C	Integer flag indicating input form of control uncertainty matrices.
IPFØRM	I	C	Integer flag indicating input form of knowledge uncertainty matrices.
LIST	O	C	Array containing parameter numbers for all parameters in augmented state vector.
LISTDY	O	C	Dynamic parameter augmentation numbers.
LØCAUG	O	C	Array locating sub-blocks within augmented covariance. (See AUGCNV, Section 3.3.1).
LØCBLK	O	C	Array locating covariance sub-blocks within block form (See AUGCNV, Section 3.3.1).

Variable	Input/ Output	Argument/ Common	Definition
LØCLAB	O	C	Array locating state vector partitions within LIST and AUGLAB arrays.
LØCFØ	O	C	Location in blank common where TRAJ (Section 3.5) picks up covariance matrix to be integrated.
LØCS	O	C	Location in blank common of areas available to TRAJ (Section 3.5).
MAXAUG	I	C	Maximum allowable length of augmented state vector.
MAXDIM	I	C	Array of maximum allowable dimensions of individual state vector partitions.
NAUG	O	C	Length of augmented state vector.
NAUGSQ	O	C	NAUG*NAUG.
NBLK	O	C	Number of words occupied by augmented covariance stored in block form (See AUGCNV, Section 3.3.1).
NDIM	O	C	Array of current dimensions of individual augmented state vector partitions.
NEPHEL	O	C	Number of ephemeris elements in augmented state vector.
NGUID	I	C	Number of guidance events to be executed.
NPHSTM	O	C	Number of dynamic parameters included in transition matrices on STM file.
NSØLVE	O	C	Total number of parameters to be solved-for by filter (including S/C state).

Variable	Input/ Output	Argument/ Common	Definition
P	O	C	Location in blank common of knowledge covariance.
PDØT	I	C	Logical flag for covariance propagation. =T, integrate covariance =F, use state transition matrices.
PG	O	C	Location in blank common of control covariance.
PG1	O	C	Location in blank common of NAUG X NAUG storage blocks used for guidance.
PG2	O	C	
PG3	O	C	
PG4	O	C	
PHI	O	C	Location in blank common of transition matrix.
PLØCAL	O	C	Working locations in blank common for intermediate operations on covariances and transition matrices.
PTEMP	O	C	
PWLS	O	C	Location in blank common of weighted least squares reference covariance.
XLAB	I	C	Array of Hollerith labels for all parameters available for augmentation.

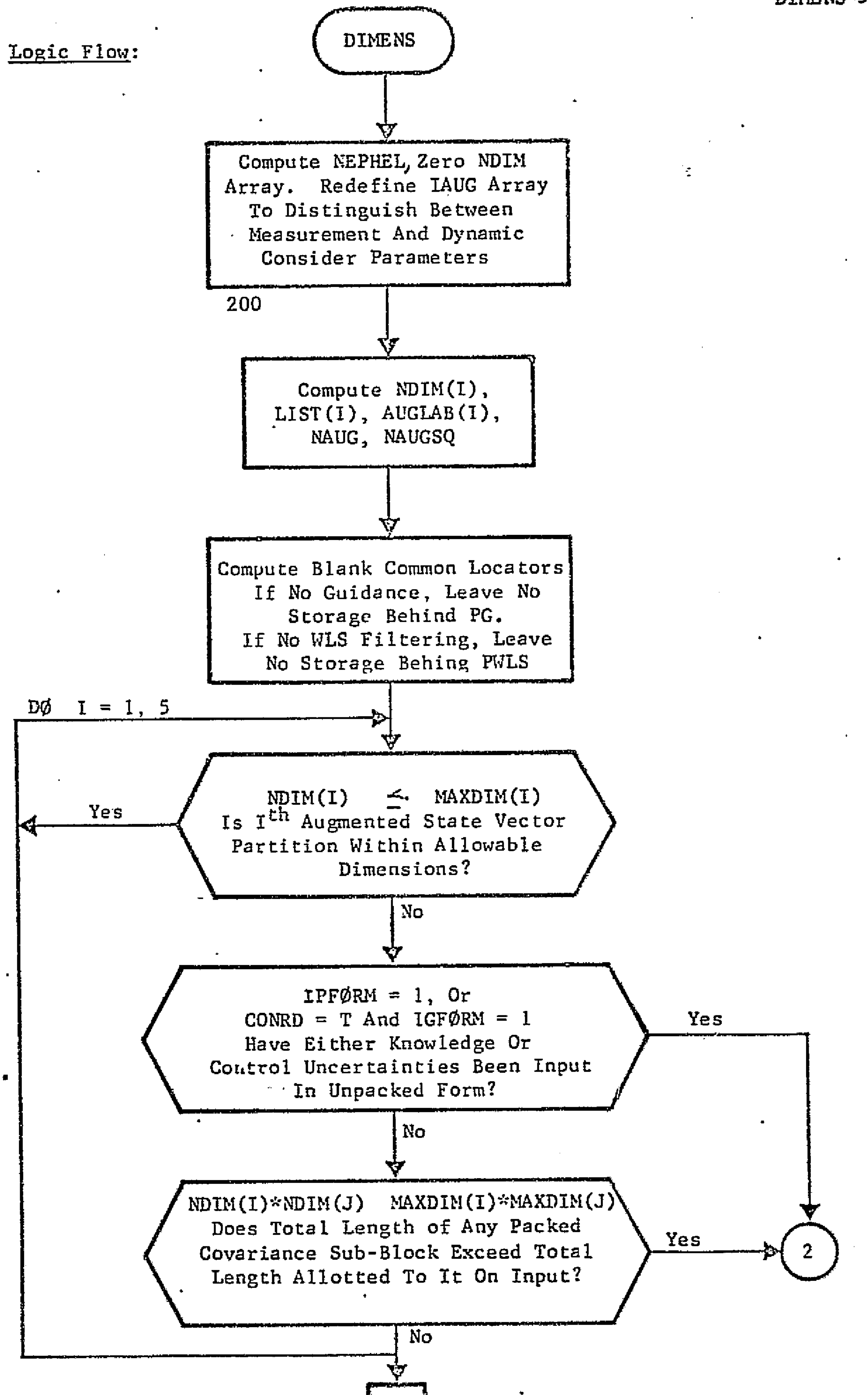
Local Variables: None

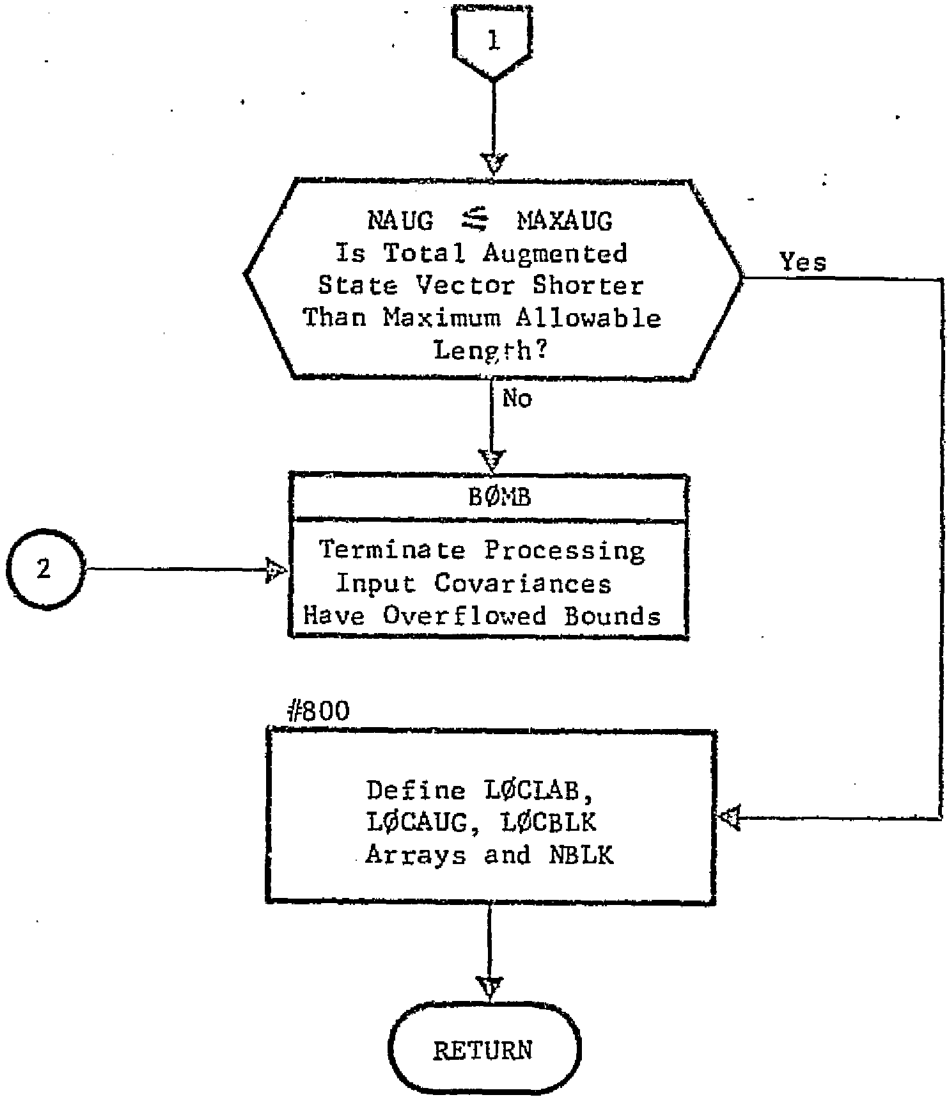
Subroutines Called: BØMB

Calling Subroutines: INPUTG

Common Blocks: WØRK, (BLANK), DATAGI, DATAGR, DIMENS, LABEL, LØCATE, LØGIC, MEASI, SCHEDI, TRAJ2

Logic Flow:


DIMENS
Compute NEPHEL, Zero NDIM Array. Redefine IAUG Array To Distinguish Between Measurement And Dynamic Consider Parameters
200
Compute NDIM(I), LIST(I), AUGLAB(I), NAUG, NAUGSQ
Compute Blank Common Locators If No Guidance, Leave No Storage Behind PG. If No WLS Filtering, Leave No Storage Behing PWLS
DØ I = 1, 5
NDIM(I) ≤ MAXDIM(I)
Is I th Augmented State Vector Partition Within Allowable Dimensions?
Yes
No
IPFØRM = 1, Or CONRD = T And IGFØRM = 1
Have Either Knowledge Or Control Uncertainties Been Input In Unpacked Form?
Yes
No
NDIM(I)*NDIM(J) MAXDIM(I)*MAXDIM(J)
Does Total Length of Any Packed Covariance Sub-Block Exceed Total Length Allotted To It On Input?
Yes
2
No


1
NAUG ≤ MAXAUG
Is Total Augmented
State Vector Shorter
Than Maximum Allowable
Length?
Yes
No
BØMB
Terminate Processing
Input Covariances
Have Overflowed Bounds
2
#800
Define LØCLAB,
LØCAUG, LØCBLK
Arrays and NBLK
RETURN

3.3.10 Subroutine: DYNØ (T, DT, PHIMAT)

Purpose: To compute effective process noise.

Method: See Volume I, Analytical Manual, Section 6.2.

Remarks: For PDOT, DYNØ is used to modify the thrust bias and noise partitions of the augmented covariance when the number of thrusters has changed (at thrust switching events).

To change the process noise model, subroutines DYNØ, ØUTPTG, and LØADFM (in TRAJ) may be affected for PDOT, and subroutines DYNØ and STMUSE may be affected for STM usage (effective process noise).

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
T	I	A	Trajectory time at beginning of propagation interval (STM only)
DT	I	A	Interval length (days).
PHIMAT	I	A	Augmented transition matrix over propagation interval.
EPTAU	I	C	Array of process noise correlation times.
EPVAR	I	C	Array of process noise variances.
GT	I	C	Transformation matrix from magnitude, cone, clock to ecliptic cartesian coordinate system evaluated at end of propagation interval.
GTSAVE	I	C	Same as GT matrix, only evaluated at beginning of propagation interval

Variable	Input/ Output	Argument/ Common	Definition
NAUG	I	C	Length of augmented state vector.
NTPHAS	I	C	Number of current thrust phase.
P	I	C	Location in blank common of knowledge covariance.
PTEMP	I	C	Location in blank common of temporary covariance.
Q	O	C	Effective process noise matrix (6x6).
THRUST	I	C	Array of thrust phase definition parameters.
TM	I	C	Conversion constant, seconds/day.

Local Variables:

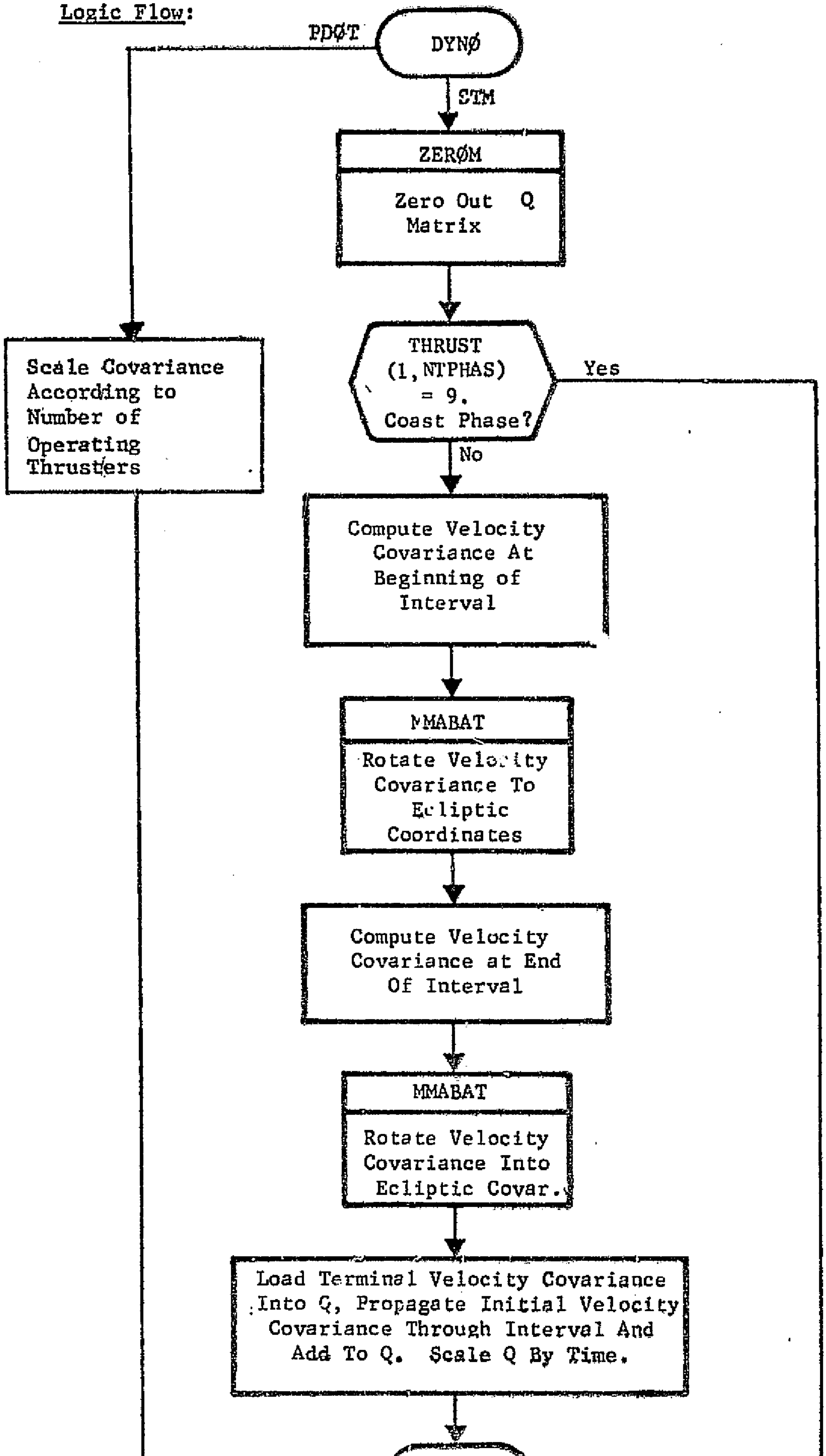
Variable	Definition
NCPHAS	Number of next thrust phase
ØMECØV	Effective velocity covariance in magnitude, cone, clock coordinates.
PHISUB	6x3 sub-block of PHIMAT representing sensitivity of position and velocity at end of interval to velocity at beginning of interval.
THRSTR	Ratio of operating thruster at phase change.
VEFF1	Effective ecliptic cartesian velocity covariance at beginning of interval.
VEFF2	Effective ecliptic cartesian velocity covariance at end of interval.

Subroutines Called: AMABT, LØCLST, MMABAT, MPAK, MUNPAK, SCALE, SDVAR, SYMUP, VARSD, ZERØM

Calling Subroutines: CØVP, GUIDE, SETEVN

Common Blocks: WØRK, (BLANK), CØNST, DIMENS, LØCATE, LØGIC, PRØPR, TRAJ1, TRAJ2

Logic Flow:


DYNØ
PDØT
STM
ZERØM
Zero Out Q Matrix
THRUST (1,NTPHAS) = 9. Coast Phase?
Yes
No
Scale Covariance According to Number of Operating Thrusters
Compute Velocity Covariance At Beginning of Interval
MMABAT
Rotate Velocity Covariance To Ecliptic Coordinates
Compute Velocity Covariance at End Of Interval
MMABAT
Rotate Velocity Covariance Into Ecliptic Covar.
Load Terminal Velocity Covariance Into Q, Propagate Initial Velocity Covariance Through Interval And Add To Q. Scale Q By Time.

3.3.11 Subroutine: EIGPRN (A, N, PVSUB, PZERØ, VZERØ)

Purpose: To compute and print eigenvectors and eigenvalues of an input matrix.

Remarks: Two options on computing eigenvalues and vectors are provided. The first operates on the complete input matrix. The second operates on the 3x3 position and velocity sub-blocks only, which are assumed to be the first and second 3x3 diagonal sub-blocks, respectively.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
A	I	A	Input matrix.
N	I	A	Dimension of input matrix (assumed to be square).
PVSUB	I	A	Logical flag controlling computation option. =T, operate on position and velocity sub-blocks. =F, operate on complete matrix.
PZERØ	I	A	Off-diagonal annihilation value for complete matrix if PVSUB = .FALSE. or for position sub-block only if PVSUB = .TRUE.
VZERØ	I	A	Off-diagonal annihilation value for velocity sub-block if PVSUB = .TRUE. Not used if PVSUB = .FALSE.

Local Variables:

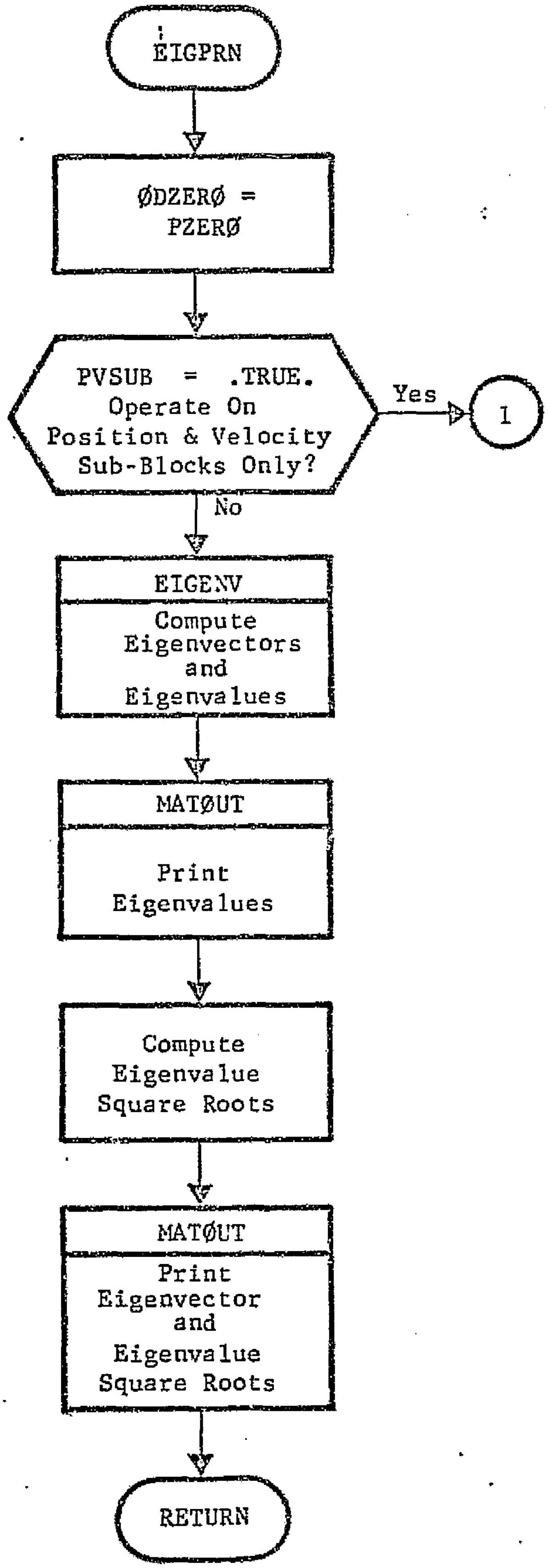
Variable	Definition
ICYCLE	Cycle control flag when PVSUB = .TRUE. indicating whether processing position or velocity sub-block.
ØDZERØ	Off-diagonal annihilation value given to EIGENV.
VALPV	Array of eigenvalues returned by EIGENV.
VECPV	Array of eigenvectors returned by EIGENV.

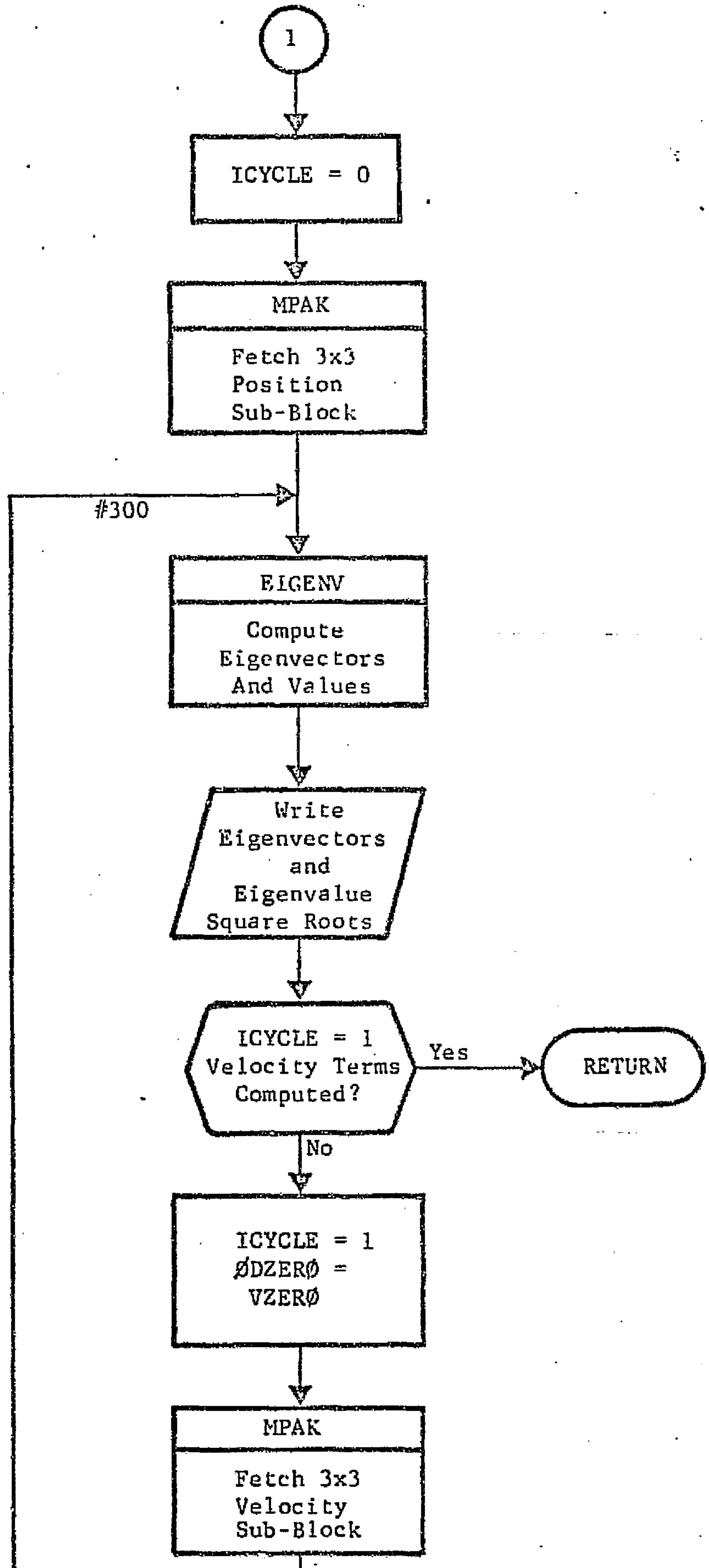
Subroutines Called: EIGENV, MATØUT, SQRT, MPAK

Calling Subroutines: SETEVN, RELCØV

Common Blocks: None

Logic Flow:


EIGPRN
ØDZERØ = PZERØ
PVSUB = .TRUE.
Operate On
Position & Velocity
Sub-Blocks Only?
Yes
1
No
EIGENV
Compute
Eigenvectors
and
Eigenvalues
MATØUT
Print
Eigenvalues
Compute
Eigenvalue
Square Roots
MATØUT
Print
Eigenvector
and
Eigenvalue
Square Roots
RETURN


1
ICYCLE = 0
MPAK
Fetch 3x3 Position Sub-Block
#300
EIGENV
Compute Eigenvectors And Values
Write Eigenvectors and Eigenvalue Square Roots
ICYCLE = 1 Velocity Terms Computed?
Yes
RETURN
No
ICYCLE = 1
ØDZERØ = VZERØ
MPAK
Fetch 3x3 Velocity Sub-Block

3.3.12 Subroutine: ESCHED (KIND, NCNT, NSTØP, TIME)

Purpose: To modify event counters to guarantee that of all events requested in namelist $GØDSEP, only those occurring between the initial and final times of the present error analysis are scheduled.

Method: If five events of a single type are scheduled according to namelist $GØDSEP, three of which occur before trajectory time TCURR, the remaining two events are not shifted into the first two locations for that event. Rather, the event counter is set to 3, informing the scheduler that the fourth event of that type will be the first scheduled.

Remarks: If any guidance events are scheduled, but the last is not scheduled within .5 day of error analysis final time, this subroutine automatically schedules an additional guidance event of policy zero. This merely forces a print of all control uncertainties at the final time.

Also, to minimize complexity of SCHED (Section 3.3.36), guidance event times are adjusted by the delay time in this subroutine.

Input/Output:

Variable	Input/ Output	Namelist/ Common	Definition
KIND	I	A	Event code. = 2, eigenvector = 3, thrust = 4, guidance = 5, prediction
NCNT	O	A	Event counter, set equal to number of events scheduled by namelist $GØDSEP which must be skipped during execution.
NSTØP	I/O	A	Total number of events of type KIND, including those skipped according to NCNT.
TIME	I	A	Array of scheduled event times.
EVLAB	I	C	Array of Hollerith event labels.
IGPØL	I	C	Array of guidance policy flags.
IGREAD	I	C	Array of guidance namelist read control flags.
TCURR	I	C	Current (and initial) trajectory time.
TCUTØF	I	C	Array of guidance event cutoff times.
TDELAY	I	C	Array of guidance event delay times.
TFINAL	I	C	Trajectory final time.
TPRED2	I	C	Array of times predicted to

Local Variables:

Variable	Definition
NUMBER	Actual number of events of code KIND to be executed.

Subroutines Called: None

Calling Subroutine: ØUTPTG

Common Blocks: LABEL, SCHEDI, SCHEDR

Logic Flow: None

3.3.13A Subroutine: ESLE (P, N)

Purpose: To load equivalent station location errors into augmented covariance matrix.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
P	I/O	A	Augmented covariance matrix still in standard deviations and correlation coefficients.
N	I	A	Dimension of augmented covariance matrix.
CØRLØN	I	C	Station-to-Station longitude correlation coefficient.
IAUG	I	C	Parameter augmentation list.
IAUGST	I	C	Location of station location parameter flags in IAUG array.
LIST	I	C	List of parameters contained in augmented state vector.
NST	I	C	Number of tracking stations.
SIGLØN	I	C	Standard deviation in station longitude.
SIGRS	I	C	Standard deviation in station spin radius.
SIGZ	I	C	Standard deviation in station z-height.
STALØC	I	C	Array of station cylindrical coordinates.

Local Variables:

Variable	Definition
EQSLE	Local array equivalenced to station location standard deviation terms. EQSLE (1) = SIGRS (2) = SIGLØN (3) = SIGZ (4) = CØRLØN
ILØC	Counter for number of stations whose location uncertainties are included in the augmented state.
LØCATE	Array used to locate off diagonal positions where longitude correlations must be loaded if more than one station's location errors are augmented.

Subroutines Called: None

Calling Subroutines: INPUTG

Common Blocks: WØRK, DATAGI, DATAGR, DIMENS, MEASI, MEASR

Logic Flow: None

3.3.13B Subroutine: FBURN (SMAT, UMAX, NTARG, NCØN, CØNWT, TARWT, TECØV, GAMMA, VMAT, BURNP, LTARG, LABS, LABCØN, VTA, NAUG, TBURN, LPØN)

Purpose: To compute the low thrust guidance matrix and associated guidance parameters.

Method: See Analytic Manual, Section 6.6 (Guidance)

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
AUGLAB	I	C	Vector of labels for augmented state.
BURNP	I	A	Mass and thrust acceleration at guidance epoch and cutoff.
CØNWT	I	A	Control parameter weights.
ENGINE(10)	O	A	Exhaust velocity.
GAMMA	O	A	Guidance matrix.
LABCØN	I	A	Vector of control parameter labels.
LABS	I	A	Vector of printout labels.
LPØN	I	C	Location in blank common of knowledge covariance.
LTARG	I	A	Vector of target lables.
NAUG	I	A	Dimension of augmented state.
NCØN	I	A	Number of control parameters.
NTARG	I	A	Number of target parameters.
PTEMP	I	C	Location in blank common of temporary (working) covariance.
SMAT	I	A	Sensitivity matrix of target WRT control parameters.

Variable	Input/ Output	Argument/ Common	Definition
TARWT	I	A	Target parameter weights.
TBURN	I	A	Duration of guidance burn.
TECØV	I	A	Target error covariance before guidance.
UMAX	I	A	Vector of maximum control corrections allowed.
VMAT	I	A	Variation matrix of target WRT state (at guidance epoch).
VTA	I	A	Logical flag for variable time of arrival guidance.

Local Variables:

Variable	Definition
CGAM	Guidance matrix for constrained control parameters.
CSWATE	Weighting factor for time parameters.
DCØN	Scaling factor.
GAMT	Guidance matrix transpose used as working array.
LCØN	Local vector of control labels (LABCØN).
LISTC	Vector of control parameter numbers (new ordering).
LISTU	Vector of control parameters numbers (old ordering).
NCU	Number of constrained controls.
NUN	Number of unconstrained controls.
STEMP	Local sensitivity matrix (SMAT).
TRCØV	Target error covariance resulting from residual (non-removeable) control error.

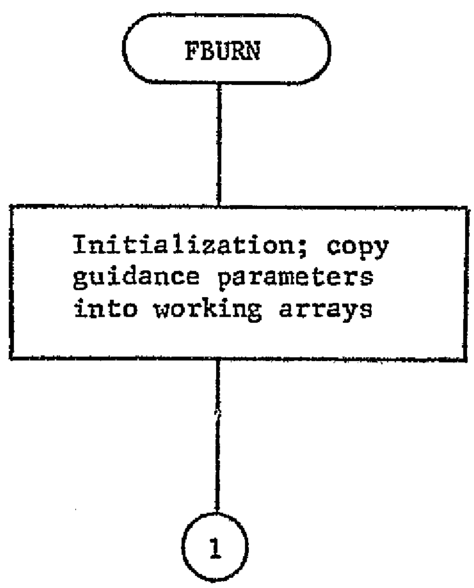
Variable	Definition
U	Control parameter correction matrix.
UMAXI	Local vector of control bounds (UMAX).
UWATE	Local vector of control weights (CØNWT).

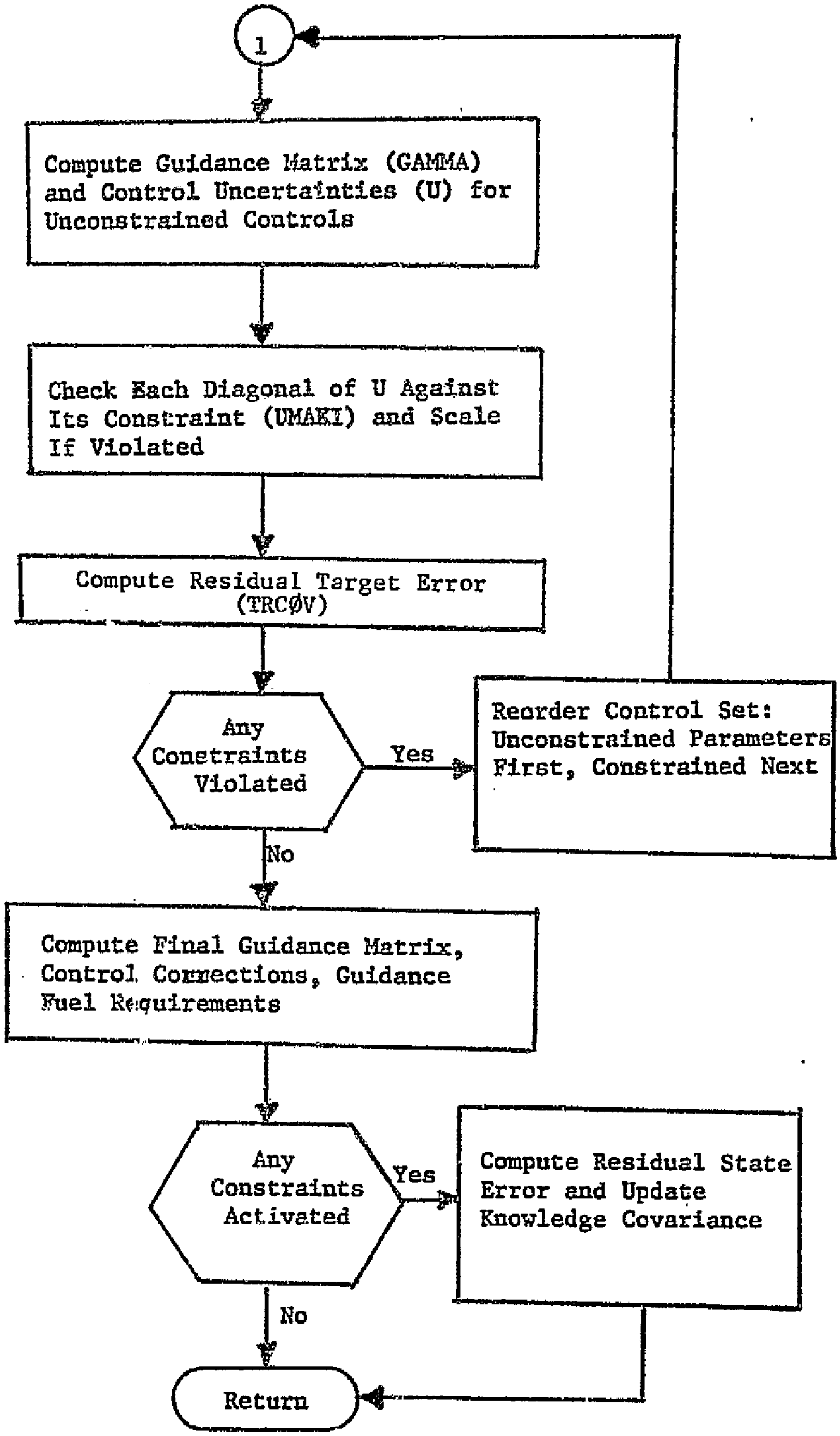
Subroutines Called: ADD, AMAB, AMABT, CØPY, CØPYT, GENINV, ICØPY, IDENT, LØADRC, MATØUT, MMABT, MMATBA, NEGMAT, PRSDEV, SCALE, VARSD, ZERØM

Calling Subroutine: GUIDE

Common Blocks: (BLANK), CONST, LABEL, LØCATE, TRAJ1, WØRK

Logic Flow:




1
Compute Guidance Matrix (GAMMA) and Control Uncertainties (U) for Unconstrained Controls
Check Each Diagonal of U Against Its Constraint (UMAXI) and Scale If Violated
Compute Residual Target Error (TRCØV)
Any Constraints Violated
Yes
Reorder Control Set: Unconstrained Parameters First, Constrained Next
No
Compute Final Guidance Matrix, Control Corrections, Guidance Fuel Requirements
Any Constraints Activated
Yes
Compute Residual State Error and Update Knowledge Covariance
No
Return

3.3.14 Subroutine: FILTR (P, PCØN, H, R, N, NS, NR, GAIN, RESID, PP)

Entry Point: FILTR2

Purpose: To compute the orbit determination filter gain for a measurement and update the knowledge covariance using that gain.

Method: A general purpose filtering routine (See Analytic Manual, Sections 6.4 and 6.5) which nominally computes the Kalman-Schmidt (KS) gain and updates the knowledge covariance. Alternately, via the entry point, FILTR2, the covariance can be updated with an input gain.

Remarks: Several places in FILTR computations require the use of sub-blocks of an input or intermediate matrix. Wherever possible, advantage is taken of internal storage formats so that the full matrix may be accessed using only the correct sub-block dimensions, eliminating requirement for pulling out the sub-block and storing it in an intermediate array.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
P	I	A	Knowledge covariance before measurement.
H	I	A	Observation matrix.

Variable	Input/ Output	Argument/ Common	Definition
GAIN	O	A	Gain matrix.
PP	O	A	Knowledge covariance after measurement.
N	I	A	Dimension of augmented covariance.
NR	I	A	Dimension of current measurement.
NS	I	A	Total number of variables and parameters being estimated by filter.
PCØN	I	A	Location in blank common of working storage as large as augmented covariance matrix.
R	I	A	Measurement white noise matrix.
RESID	O	A	Measurement residual matrix.

Local Variables:

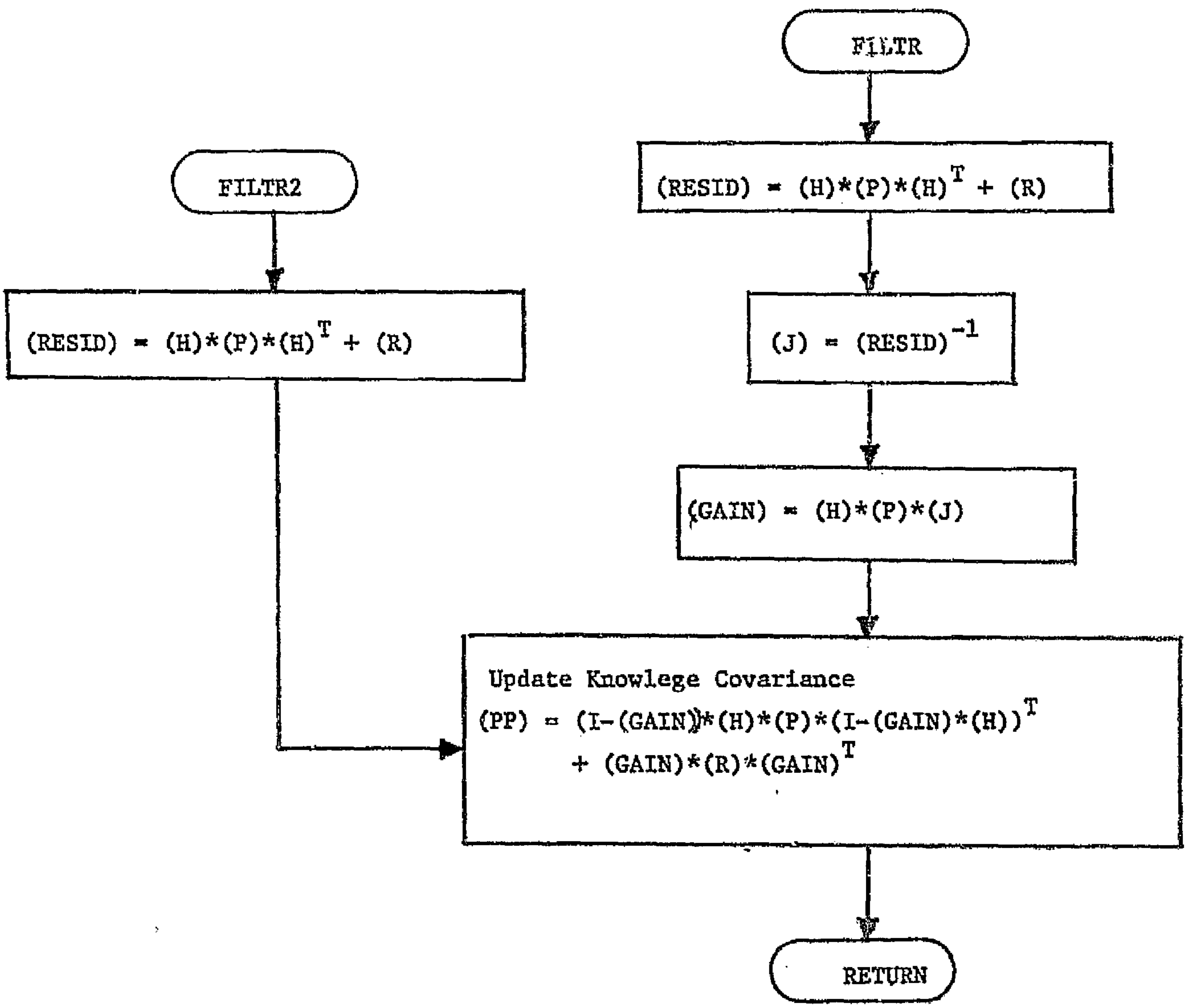
Variable	Definition
HP	Product of observation matrix and input covariance matrix.
INVRES	Location in common/WØRK/ of inverse of measurement residual matrix.
INVRS2	Location in common /WØRK/ of working storage.

Subrountines Called: AMABT, AMATBT, CØPY, INVSQM, MMAB, MMATB, SCALE, SYMTRZ

Calling Subroutines: MEAS

Common Blocks: WØRK

Logic Flow:



Pages 247 and 248 are deleted.

3.3.15 Subroutine: GAINF (K, RDWRIT)

Purpose: To read gain matrix from or write gain matrix to GAIN file (TAPE 4).

Input/Output:

Variable	Input/ Output	Namelist/ Common	Definition
K	I/O	A	Gain matrix (real).
RDWRIT	I	A	Read/write control flag = 4HREAD, read gain matrix = 5HWRITE, write gain matrix.
CHEKPR(4)	I	C	Logical check print flag, operative for both read and write modes. = T, print gain matrix to output = F, do not print gain matrix.
MESEVN	I	C	Measurement code corresponding to gain matrix.
NR	I	C	Number of columns in gain matrix.
NSØLVE	I	C	Number of rows in gain matrix.

Local Variables:

Variable	Definition
MEV	Measurement code read from GAIN file. MEV is compared to MESEVN, the code provided from SCHED (Section 3.3.36) to guarantee proper meshing of gain with its original data type.

Subroutines Called: MATØUT, BØMB

<u>Calling Subroutine</u>: MEAS

<u>Common Blocks</u>: LØGIC, MEASI, SCHEDI

<u>Logic Flow</u>: None

3.3.16 Subroutine: GAINUS (K)

Purpose: To be replaced by user if any gain matrix algorithm is desired other than Kalman-Schmidt, sequential weighted least squares, or read from GAIN file.

Remarks: Users-supplied gain is expected to be an infrequently exercised option. The user who wishes to incorporate his own algorithm should be very familiar with filtering theory. Though there are no "wrong" algorithms, any algorithm not carefully thought out -- and many that are -- will generally be meaningless and harmful. The only absolute rule is that the gain matrix has dimensions NSØLVE by NR (common/MEASI/).

Calling Subroutine: MEAS

.3.17 Subroutine: GUIDE.

Purpose: To perform all computations and printout for the execution of a guidance maneuver.

Method: Both low thrust and impulsive ΔV guidance are available. See Vol. I, Analytical Manual, Sec. 6.6 for details.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
AUGLAB	I	C	Hollerith label array for all augmented parameters.
CHEKPR(5)	I	C	Check print flag =T, low thrust guidance - print, knowledge and control uncertainties at end of burn interval and transition matrix over burn interval. ΔV guidance - prints eigenvalues and eigenvectors of ΔV covariance. =F, no optional print
CØNWT	I/O	C	Array of control weights.
DELAY	I	C	Guidance delay time for current maneuver.
DXDKBR	I	C	Keplerian to cartesian coordinate transformation for ephemeris body at beginning of burn interval.
DYNØIS	I	C	Dynamic noise flag
EØV	I	C	Velocity covariance off-diagonal annihilation value for eigenvalue/vector computation.
GT	I/O	C	Transformation matrix for dynamic noise computation.
GTBURN	I	C	GT matrix evaluated at beginning of burn interval.

Variable	Input/ Output	Argument/ Common	Definition
GTDLAY	I	C	GT matrix evaluated at beginning of guidance delay period.
GTSAVE	I/O	C	Transformation matrix for dynamic noise computation.
IEP	I	C	See UREL, VREL below.
IPØL	I	C	Guidance policy for current maneuver.
IREAD	I	C	Namelist $GEVENT read control flag for current maneuver.
ITP	I	C	See UREL, VREL below.
LØCTC	I	C	Location in blank common of transition matrix from cutoff time to target condition time.
NAUG	I	C	Length of augmented state vector.
NCNTG	I	C	Number of current guidance maneuver.
NCØN	I	C	Number of low thrust controls.
NEPHEL	I	C	Number of ephemeris elements in augmented state.
NPHSTM	I	C	Dimension of state transition matrix from TRAJ (Sec. 3.5) with dynamic parameters only.
NTP	I	C	Code number for target body.
P	I	C	Location in blank common of knowledge covariance at beginning of guidance delay period.
PG	I	C	Location in blank common of control covariance at beginning of guidance delay time.
PG1	I	C	Locations in blank common for intermediate covariances required for guidance computations.
PG2	I	C	(PG1–PG4, bracketed together)
PG3	I	C	
PG4	I	C	

Variable	Input/ Output	Argument/ Common	Definition
PHI	I	C	Location in blank common of transition matrix over delay period.
PI	I	C	Mathematical constant, π
PLAB	I	C	Array of knowledge covariance labels.
PLØCAL	I	C	Location in blank common of covariance-sized working storage.
PTEMP	I	C	Same as PLØCAL.
RAD	I	C	Conversion constant, degrees/ radian.
S	I	C	Sensitivity matrix, cutoff state w.r.t. controls.
SMAT	I/O	C	Sensitivity matrix, targets WRT controls.
TBURN	I	C	Burn interval duration for current maneuver.
TIMFTA	I	C	Target condition evaluation time for fixed time of arrival guidance.
TM	I	C	Conversion constant, seconds/day.
TØFF	I	C	Cutoff time for current maneuver.
TØN	I	C	Startup time for current maneuver.
TSTM	I	C	Most recent STM file time point.
TSTØP	I	C	Trajectory stop time from integrator for B-plane or closest approach targeting.
UREL(1,IEP)	I	C	S/C position relative to ephemeris body at target condition time.
VARDV	I	C	Array of execution error variances.
VARMAT	I/O	C	Variation matrix, sensitivity of target conditions to cutoff state.
VREL(1,IEP)	I	C	S/C velocity relative to ephemeris body at target condition time.

Variable	Input/ Output	Argument/ Common	Definition
VRNIER	I	C	Logical flag = T, current maneuver is vernier. = F, current maneuver not vernier.

Local Variables:

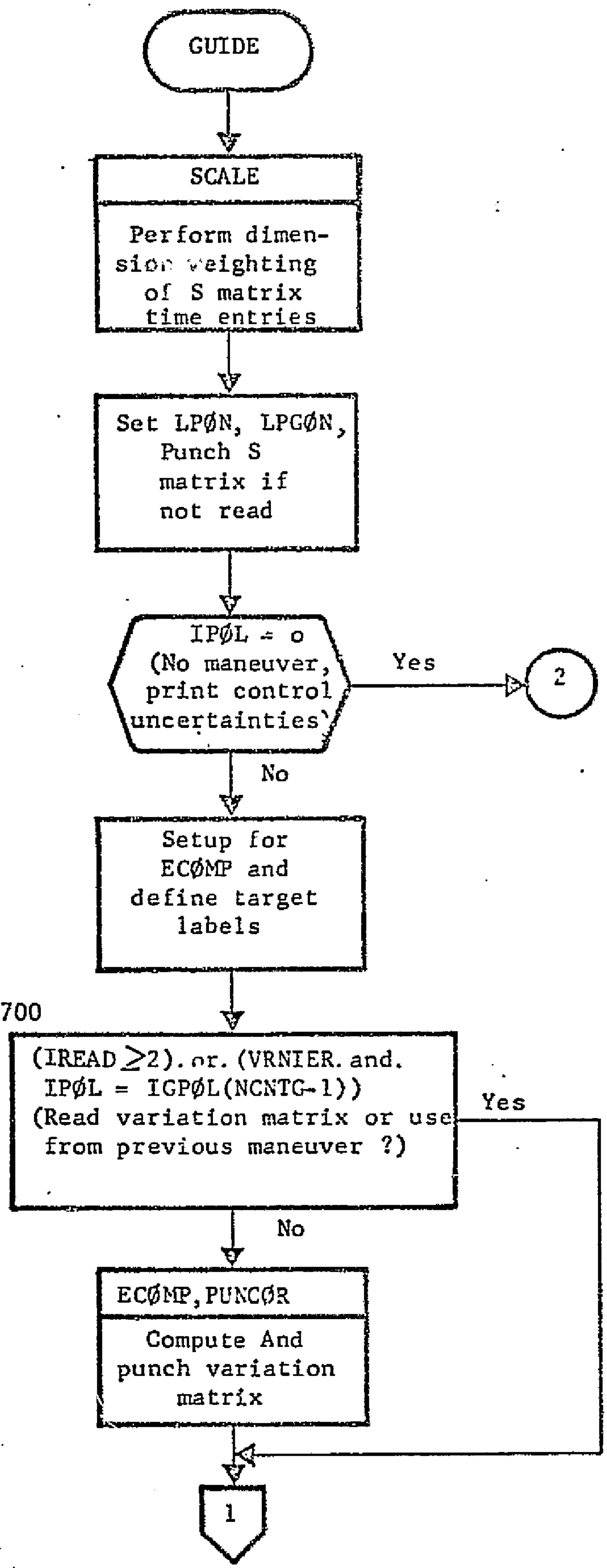
Variable	Definition
LABCØN	Array of control Hollerith labels.
CSWATE	Dimensional weighting for start-up and cutoff time controls.
DELTAV	Expected velocity update for ΔV guidance.
DVCØV	Impulsive ΔV covariance
DVM	Mean ΔV magnitude.
ETA	Variation matrix, target conditions wrt state at target condition time.
GAMMA	Guidance matrix
LABS	Labelling array
ITARG	Input parameter to ECØMP (Sec. 3.6.5)
JSTØP	Input parameter to ECØMP (Sec. 3.6.5)
LPGØFF	Location in blank common of control covariance at cutoff time.
LPGØN	Location in blank common of control covariance at startup time.
LPØFF	Location in blank common of knowledge covariance at cutoff time.
LPØN	Location in blank common of knowledge covariance at startup time.
NTARG	Number of targets.
PHIBRN	6 x 6 state transition matrix over burn interval.
PHITAR	6 x 6 state transition matrix from cutoff to target condition time.

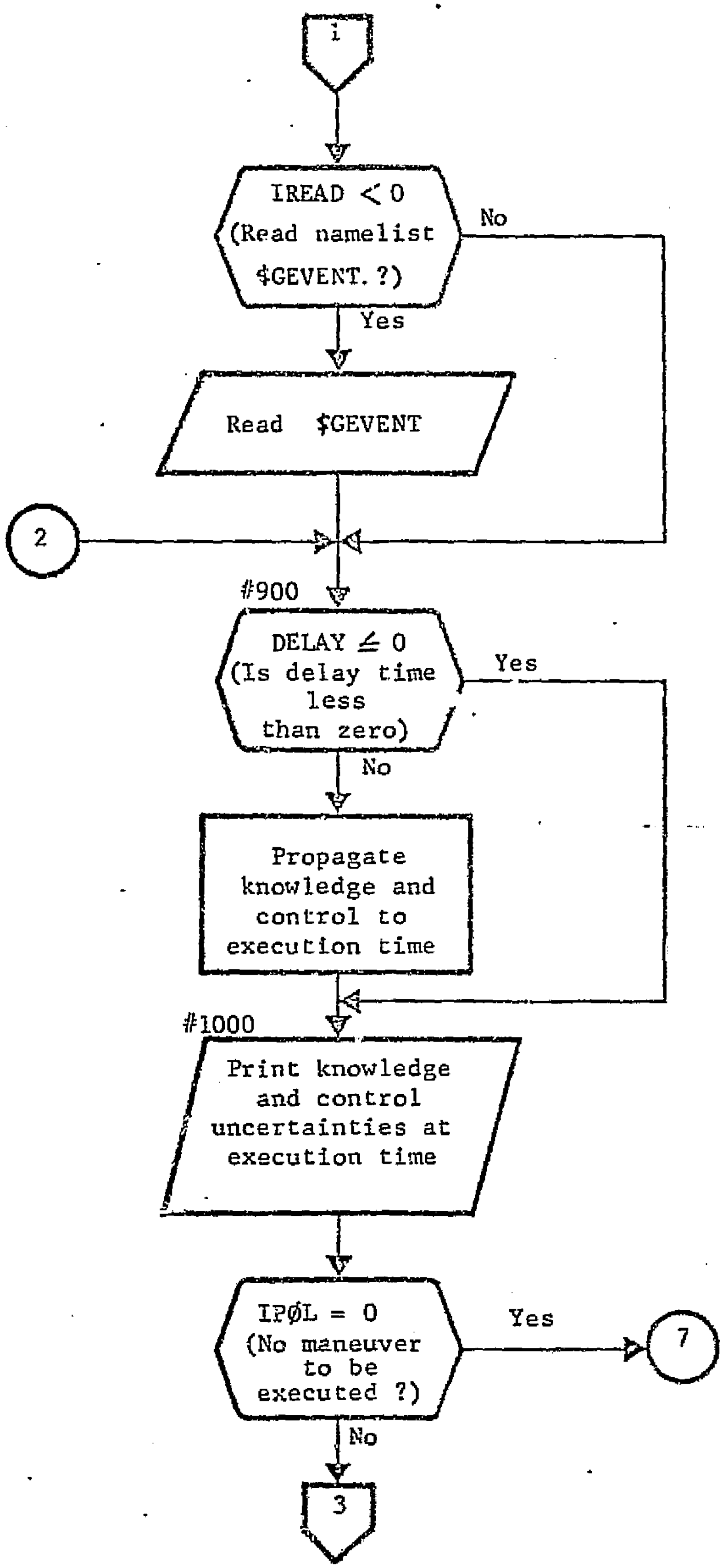
Variable	Definition
SIGDV	Standard deviation in ΔV.
LABTAR	Array of target labels.
TARTIM	Target condition evaluation time.
TEMP	Hollerith prefix.
LTARG	Current target label.
TRS	Trace of ΔV covariance.
VMAT	Variation matrix, target parameters WRT state at guidance epoch.
VTA	Logical flag for variable time of arrival low thrust XYZ guidance (if TRUE).

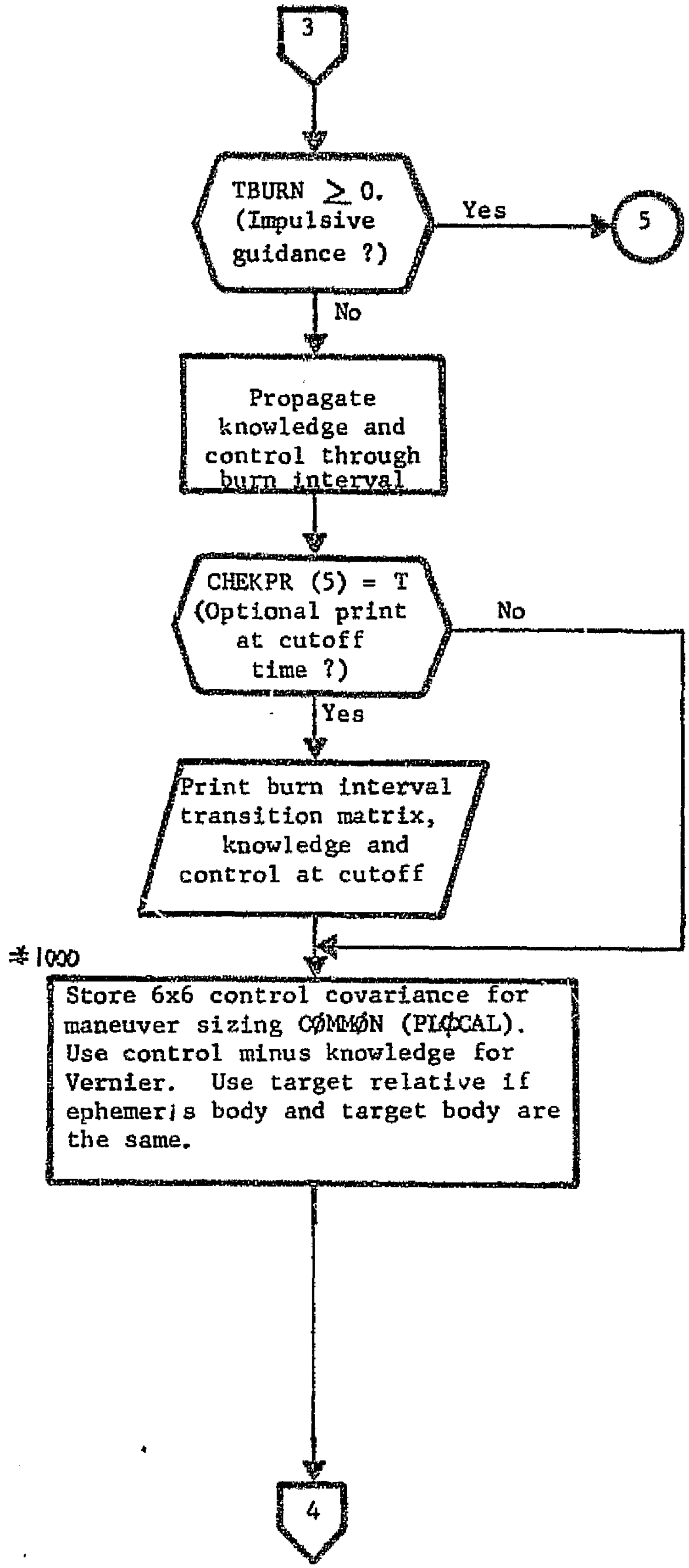
Subroutines Called: ADD, CØPY, CØRREL, DYNØ, ECØMP, EIGENV, FBURN, GENINV, ICØPY, MATØUT, MMAB, MMABAT, MPAK, MUNPAK, NEGMAT, PRØP, PRSDEV, PUNCØR, RELCØV, SCALE, SUB, VARSD, VERR.

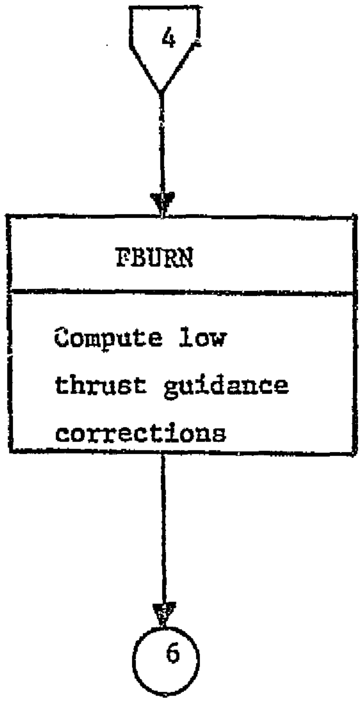
Common Blocks: WØRK, (BLANK), CØNST, DIMENS, GUIDE, KEPCØN, LABEL, LØCATE, LØGIC, MEASI, PRØPR, SCHEDI, SCHEDR, TIME, TRAJ1, TRAJ2.

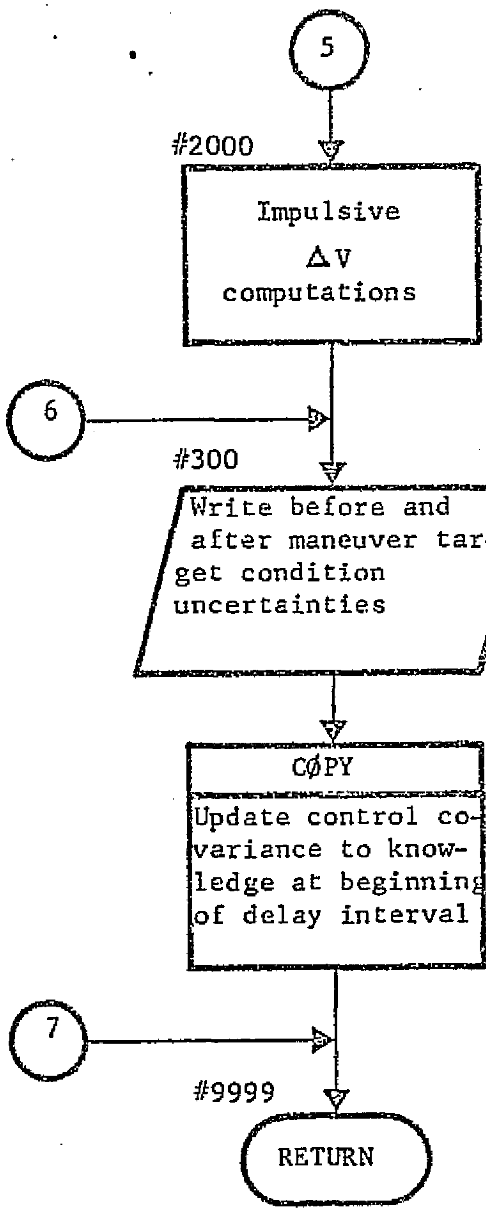
Logic Flow:


GUIDE
SCALE
Perform dimension weighting of S matrix time entries
Set LPØN, LPGØN, Punch S matrix if not read
IPØL = o (No maneuver, print control uncertainties)
Yes
2
No
Setup for ECØMP and define target labels
#700
(IREAD ≥2).or.(VRNIER.and. IPØL = IGPØL(NCNTG-1))
(Read variation matrix or use from previous maneuver ?)
Yes
No
ECØMP,PUNCØR
Compute And punch variation matrix
1


i
IREAD < 0
(Read namelist
$GEVENT. ?)
No
Yes
Read $GEVENT
2
#900
DELAY ≤ 0
(Is delay time less than zero)
Yes
No
Propagate knowledge and control to execution time
#1000
Print knowledge and control uncertainties at execution time
IPØL = 0
(No maneuver to be executed ?)
Yes
7
No
3


3
TBURN ≥ 0.
(Impulsive guidance ?)
Yes
5
No
Propagate knowledge and control through burn interval
CHEKPR (5) = T
(Optional print at cutoff time ?)
No
Yes
Print burn interval transition matrix, knowledge and control at cutoff
#1000
Store 6x6 control covariance for maneuver sizing CØMMØN (PLØCAL). Use control minus knowledge for Vernier. Use target relative if ephemeris body and target body are the same.
4


4
FBURN
Compute low
thrust guidance
corrections
6


5
#2000
Impulsive
ΔV
computations
6
#300
Write before and
after maneuver tar-
get condition
uncertainties
CØPY
Update control co-
variance to know-
ledge at beginning
of delay interval
7
#9999
RETURN

3.3.18 Subroutine: INPUTG

Purpose: To control all inputs to GØDSEP

Remarks: Common/LØCAL/ appears in this subroutine only and is an ordering artifice to equivalence its elements to the array LØCATE.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
CØNRD	Ø	C	Logical flag =T, control uncertainties read in =F, control uncertainties not read in
IEPHEM	Ø	C	Flag indicating coordinate system of ephemeris elements.
IGFØRM	Ø	C	Flag indicating form of input control uncertainties.
IPFØRM	Ø	C	Flag indicating form of input knowledge uncertainties.
ISTMF	I	C	STM file usage flag
MAXDIM	I	C	Array of maximum dimensions allowed on input covariance sub-blocks.
NAUG	Ø	C	Length of augmented state vector.
P	Ø	C	Location in blank common of knowledge covariance.
PG	Ø	C	Location in blank common of control covariance
XLAB	I	C	Array of Hollerith labels for all possible augmentation parameters.

Local Variables:

Variable	Definition
CXS, CXU, CXW, PS, CSU, CSV, CSW, PU, CUV, CUW, PV, CVW, PW	Locations in blank common of input covariance matrix sub-blocks of the same name.

Variable	Definition
NTØT	Total number of words allocated for each of knowledge and control uncertainties to be read in namelist $GØDSEP.

Subroutines Called: NMLIST, DIMENS, PPAK, ESLE, SYMUP

Calling Subroutines: DATAG

Common Blocks: WØRK(BLANK), DATAGR, DATAGI, DIMENS, LØCATE, MEASI, TRAJ2, LØCAL

Logic Flow: None

3.3.19 Subroutine: LØADRC (A, MA, NA, LISTA, C, M, N, LISTC, LTRAN)

Entry Points: LØDCØL, LØDRØW

Purpose: To load selected rows or columns from one matrix to another.

Method: A list of codes (LISTA for matrix A and LISTC for matrix C) is associated with either column entries, row entries or both. The two matrix codes are compared and rows or columns having common codes are loaded from A to C.

LØDCØL uses LISTC to define the columns of C. Letting the index J run from 1 to N, for each value of J, LISTA is searched for an element JJ such that LISTC(J) = LISTA(JJ). If no equality is found, no operation is performed on column J of matrix C. If an equality is found, the elements of row JJ in matrix A are copied into row J of C.

LØDRØW functions the same way for the rows of C as LØDCØL does for columns. LISTC and LISTA are then assumed to define the rows of C and A, respectively.

LØADRC loads rows and columns simultaneously for square matrices where a single list can denote ordering for both rows and columns, such as covariance and transition matrices. For the simultaneous loading, an intermediate transformation array LTRAN is used. LTRAN(I) is zero if the Ith parameter of LISTC does not appear in LISTA, or is equal to II if LISTA(II) = LISTC(I). Individual elements are transferred from A to C by

$$C(I,J) = A(LTRAN(I)), LTRAN(J))$$

if LTRAN(I) > 0 and LTRAN(J) > 0, otherwise element C(I,J) is not changed from input value.

Remarks: The argument LTRAN is working storage and is used only when LØADRC is called. It must have a length at least as great as LISTC. The inputs NA and N are ignored for LØADRC, A is assumed to be MAxMA and C to be MxM.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
A	I	A	Input matrix.
MA	I	A	Number of rows in A.
NA	I	A	Number of columns in A.
LISTA	I	A	Vector list of code numbers for rows/columns of A.
C	O	A	Output matrix.

Variable	Input/ Output	Argument/ Common	Definition
M	I	A	Number of rows in C.
N	I	A	Number of columns in C.
LISTC	I	A	Vector list of code numbers for rows/columns of C.
LTRAN	O	A	Transformation list from A to C in LØADRC designed as working storage with no specific output function. Must have length greater than or equal to that of LISTC.

Local Variables:

Variable	Definition
MIN	LØDCØL - minimum of (M, MA) LØDRØW - minimum of (N, NA) When copying rows or columns MIN is the row or column length. It guarantees that the length of rows or columns in neither A nor C is exceeded.

Subroutines Called: None

Calling Subroutines: STMRDR, GUIDE, CØVP, PRED, STMUSE, RELCØV

Common Blocks: None

Logic Flow: None

3.3.20A **Function:** LØCLST (IPARAM)

Purpose: To locate the position of a parameter in the augmented state vector.

Input/Output:

Variable	Input/ Output	Argument Common	Definition
IPARAM	I	A	Code number of parameter to be located.
NAUG	I	C	Dimension of augmented state vector.
LIST	I	C	Vector of code numbers in augmented state.
LØCLST	O	F*	Parameter location, if in augmented state.

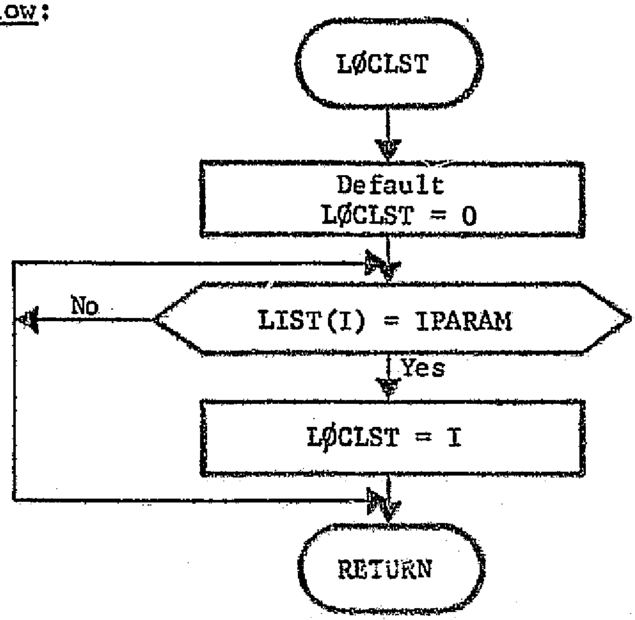
Local Variables: None

Subroutines Called: None

Calling Subroutines: ØBSERV

Common Blocks: DIMENS

Logic Flow:



*Function Value Output

3.3.20B Subroutine: MASSIG (IFLAG, P, PG, DT)

Purpose: To compute the estimated and cumulative spacecraft mass variances.

Method: See Analytic Manual, Section 6.2 (Covariance Propagation).

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
DT	I	A	Propagation interval.
ENGINE(10)	I	C	Exhaust velocity.
EPTAU	I	C	Thrust noise correlation times.
EPVAR	I	C	Thrust noise variances.
IAUGDC	I	C	Vector of flags for dynamic parameters.
IFLAG	I	A	Flag for computational control. = 0, do not average acceleration. = 1, initialize SAVACC. = 2, update mass variance. = 3, update and print mass variance.
NAUG	I	C	Dimension of augmented state.
NTPHAS	I	C	Current thrust phase number.
P	I	A	Knowlege covariance.
PG	I	A	Control covariance.
SAVACC	I/O	C	Previous thrust acceleration.
SCMASS	I	C	Current S/C mass.
SCMVAR	I/O	C	Current mass variance.
THRACC	I	C	Thrust acceleration vector.
THRUST	I	C	S/C thrust array.

Local Variables:

Variable	Definition
FLØW	S/C mass flow rate.
INITA	Initialization flag = 0, do not average acceleration. = 1, use average acceleration.
TAMAG	Thrust acceleration magnitude.

Subroutines Called: CØPY, LØCLST, VECMAG

Calling Subroutines: GØDSEP, SETEVN

Common Blocks: CØNST, DIMENS, LØGIC, PRØPR, TRAJ1, TRAJ2, WØRK

Logic Flow: None.

3.3.21 Program: MEAS

Purpose: Executive control for measurement processing.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
GAIN	Ø	C	Location in blank common of gain matrix.
H	Ø	C	Location in blank common of observation matrix.
IDATYP	Ø	C	See ØBSERV, 3.3.26.
IGAIN	I	C	Gain matrix flag. = 1, Kalman-Schmidt (KS) = 2, sequential weighted least squares (WLS). = 3, user-supplied. = 4, read from GAIN file
ISTA3	Ø	C	See ØBSERV, 3.3.26.
NAUG	I	C	Length of augmented state vector.
NR	Ø	C	Length of measurement vector.
P	I	C	Location in blank common of knowledge covariance after measurement.
PRINT	Ø	C	Logic flag =T, full print for current measurement =F, do not give full print for current measurement.
PTEMP	I	C	Location in blank common of knowledge covariance before measurement,
PWLS	I	C	Location in blank common of WLS reference covariance.
SUMMARY	I	C	Logical flag =T, summary print for all measurements. =F, no summary print.

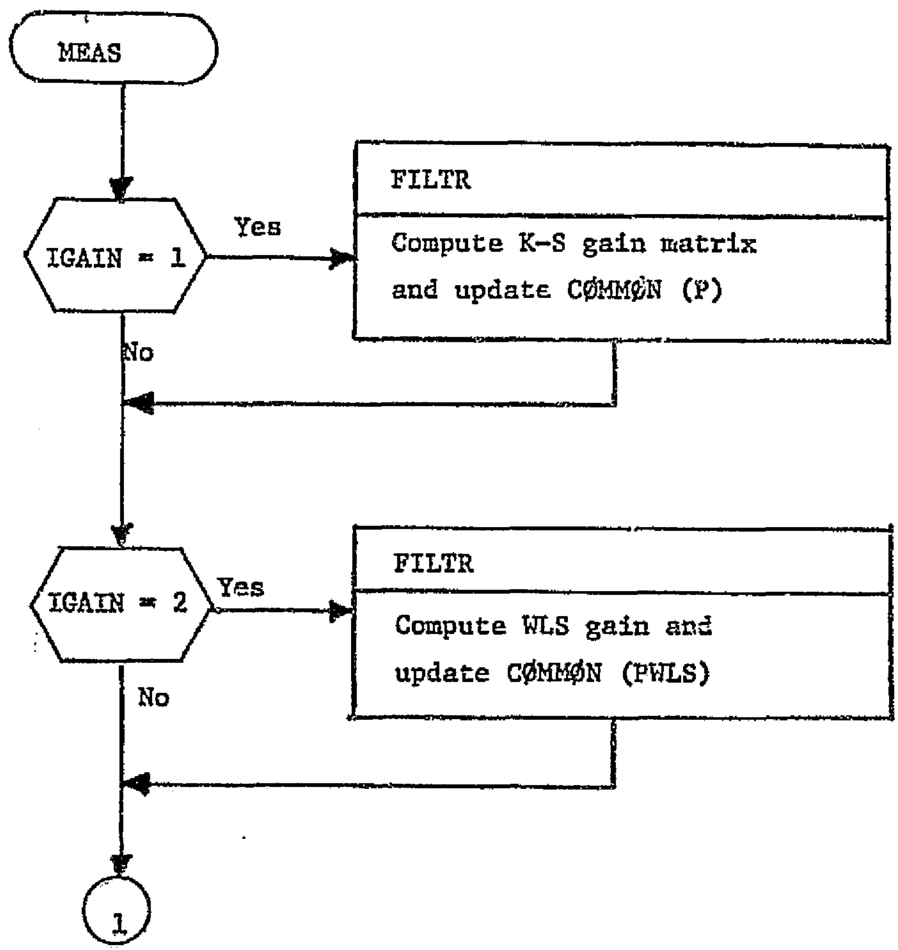
Local Variables: None

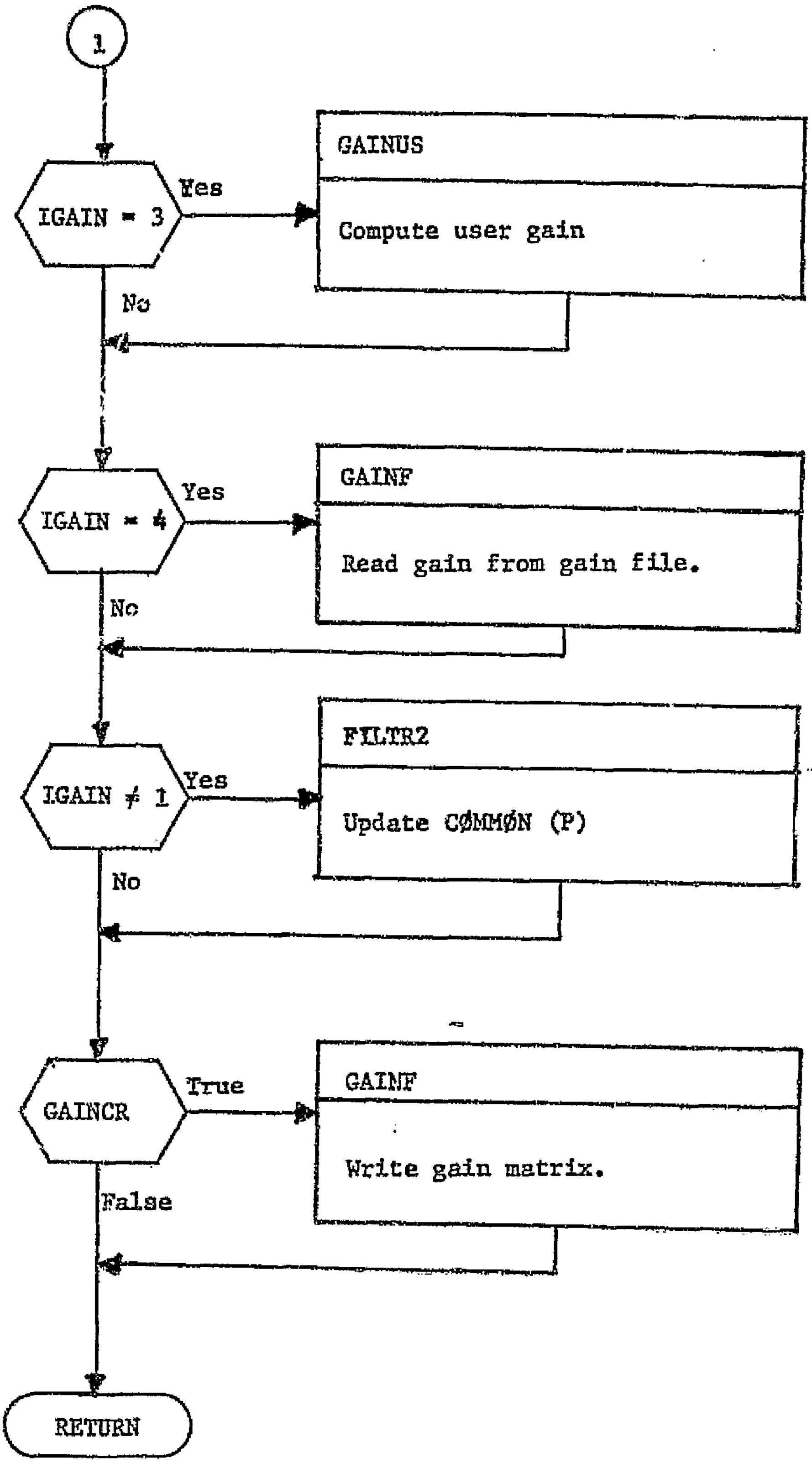
Subroutines Called: FILTR, GAINF, GAINUS MEASPR, MNØISE, ØBSERV, PCNTRL

Calling Subroutines: GØDSEP

Common Blocks: WØRK, (BLANK), DIMENS, LABEL, LØCATE, LØGIC, MEASR, MEASI

Logic Flow:




1
IGAIN = 3
Yes
No
GAINUS
Compute user gain
IGAIN = 4
Yes
No
GAINF
Read gain from gain file.
IGAIN ≠ 1
Yes
No
FILTR2
Update CØMMØN (P)
GAINCR
True
False
GAINF
Write gain matrix.
RETURN

3.3.22 Subroutine: MEASPR(TYPE)

Purpose: To control all measurement print

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
TYPE	I	A	Print type =6HBEFORE, before measurement print =5HAFTER, after measurement print
AUGLAB	I	C	Array of augmented parameter Hollerith labels.
AZMTH2	I	C	S/C azimuth angle from station ISTA2.
AZMUTH	I	C	S/C azimuth angle from station ISTA1.
CHEKPR(3)	I	C	Print covariance before and after measurement (if TRUE).
DELTIM	I	C	If >0, print transition matrices.
DXDKST	I	C	Keplerian to cartesian transformation for ephemeris elements.
ELEV	I	C	S/C elevation angle from station ISTA1
ELEV2	I	C	S/C elevation angle from station ISTA2
GAIN	I	C	Location in blank common of gain matrix.
H	I	C	Location in blank common of observation matrix.
IDATYP	I	C	General data type flag (See ØBSERV, (Section 3.3.26).
ISTA1	I	C	See ØBSERV, Section 3.3.26.
ISTA2	I	C	See ØBSERV, Section 3.3.26.
ISTA3	I	C	See ØBSERV, Section 3.3.26.
LØCLAB	I	C	Array locating state vector partitions in AUGLAB.

Variable	Input/ Output	Argument/ Common	Definition
MESEVN	I	C	Measurement code for current data type.
MESLAB	I	C	Array of measurement Hollerith labels.
NAUG	I	C	Length of augmented state vector.
NDIM	I	C	Array of lengths of individual state vector partitions.
NEPHEL	I	C	Number of ephemeris elements augmented to state.
NR	I	C	Length of current measurement vector.
NSØLVE	I	C	Total number of variables and parameters being estimated by filter.
P	I	C	Location in blank common of knowledge covariance after measurement.
PHI	I	C	Location in blank common of transition matrix.
PLAB	I	C	Array of knowledge covariance sub-block Hollerith labels.
PLØCAL	I	C	Location in blank common of covariance-sized working storage.
PRINT	I	C	Print control flag =T, full print =F, not full print
PTEMP	I	C	Location in blank common of knowledge covariance before measurement.
R	I	C	Before measurement, measurement white noise matrix; after measurement, measurement residual matrix.
SCDEC	I	C	S/C geocentric equatorial declination.

Variable	Input/ Output	Argument/ Common	Definition
SCGLØN	I	C	S/C geocentric longitude
SCMASS	I	C	S/C mass
SUMARY	I	C	Print control flag =T, summary print =F, no summary print
TCURR	I	C	Current trajectory time
TPAST	I	C	Previous trajectory time
VECLAB	I	C	Array of state vector partition Hollerith labels.

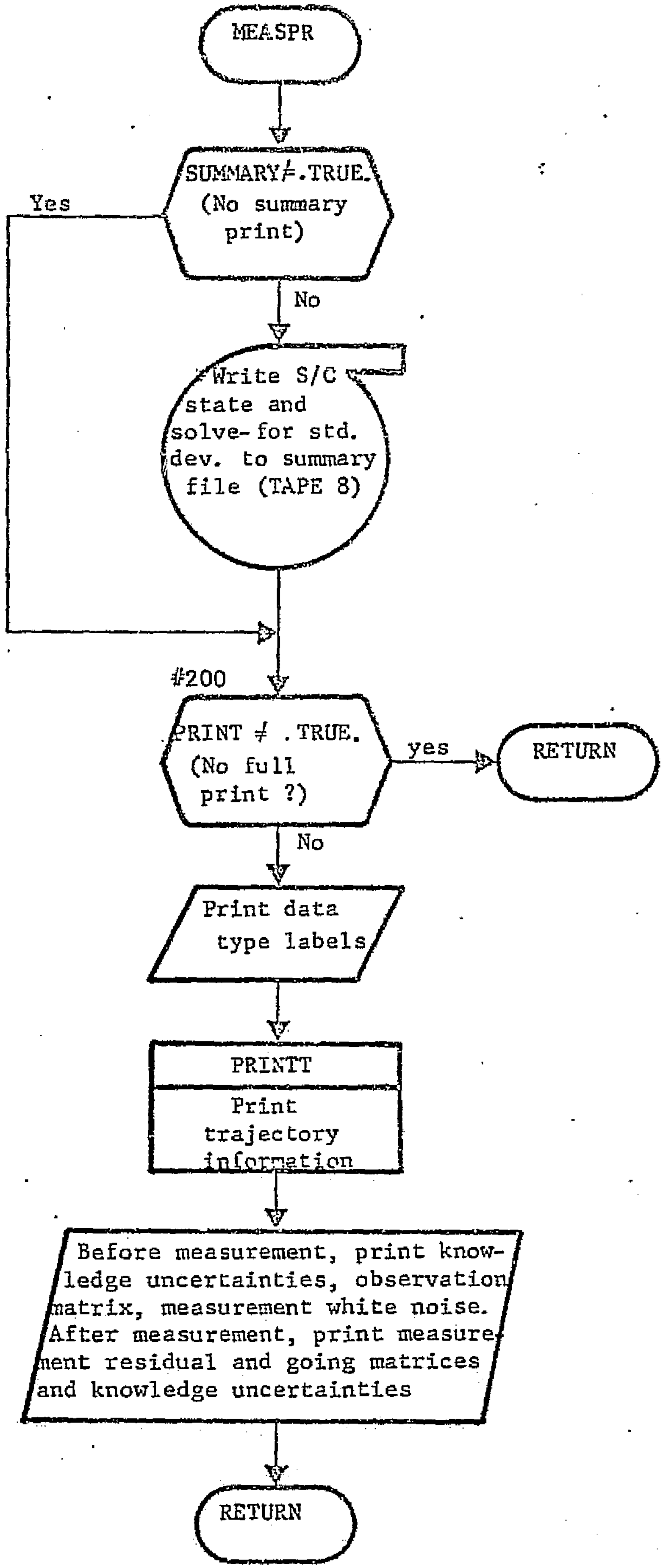
Local Variables:

Variable	Definition
AZ	"Azimuth" Hollerith label
BLANK	Hollerith "blank"
DEC	"Declination" Hollerith label
EL	"Elevation" Hollerith label
FSTA	"From Station" Hollerith label
FULPR	Flag on SUMARY print file If full print is made for current data type FULPR = 5HPRINT; otherwise FULPR = Hollerith blank.
HØLNUM	Array of Hollerith numbers
LØN	"Longitude" Hollerith label

Subroutines Called: MPAK, SQRT, JØBTLE, PRINTT, STMPR, CØRREL, RELCØV, CØPY, CØPYT, MATØUT, PRPART

Calling Subroutines: MEAS

Common Blocks: WØRK, (BLANK), DIMENS, KEPCØN, LABEL, LØCATE, LØGIC, MEASI, MEASR, SCHEDI, SCHEDR, TRAJ1, TRAJ2


MEASPR
SUMMARY ≠ .TRUE. (No summary print)
Yes
No
Write S/C state and solve-for std. dev. to summary file (TAPE 8)
#200
PRINT ≠ .TRUE. (No full print ?)
yes
RETURN
No
Print data type labels
PRINTT
Print trajectory information
Before measurement, print knowledge uncertainties, observation matrix, measurement white noise. After measurement, print measurement residual and going matrices and knowledge uncertainties
RETURN

3.3.23 Subroutine: MNØISE

Purpose: To define the measurement white noise matrix.

Method: Required elements from the measurement variance array, VARMES, are loaded into the measurement noise matrix, R.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
IDATYP	I	C	Basic data type = 1, doppler = 2, range = 3, azimuth-elevation = 4, star-planet angle = 5, apparent planet diameter. = 6, right ascension-declination.
ISTA3	I	C	Data sub-type for range and doppler. = 0, 2-way = 1, 3-way = 2, simultaneous 2-way/ 3-way = 3, differenced 2-way/ 3-way
NR	I	C	Dimension of measurement noise matrix.
R	O	C	Measurement noise matrix.
VARMES	I	C	Array of measurement white noise variances.

Local Variables: None

Subroutines Called: None

Calling Subroutines: MEAS

3.3.24 Subroutine: MSCHED

Purpose: To set up measurement and propagation event information for use by the scheduling routine SCHED (Section 3.3.36).

Remarks: If the current error analysis reads gain matrices from the gain file (generalized covariance run) all scheduling and measurement print control information will also be read from the gain file and any scheduling cards in input will be ignored. MSCHED automatically writes this information on the gain file if gain file creation has been specified in namelist $GØDSEP.

Each card read is assumed to contain four variables - START, STØP, DELT, MESCØD (for input format see GØDSEP input, Section 2.3). If the interval (START, STØP) is not completely contained in the interval (TCURR, TFINAL), the values of START and/or STØP will be adjusted so that only those events within the (TCURR, TFINAL) interval will be scheduled. Measurement events are denoted by MESCØD equal to the number of the data type, and propagation events by MESCØD equal to zero. An additional

PRECEDING PAGE BLANK NOT FILMED

option is also available to schedule measurements in any sub-interval of (TCURR, TFINAL). When any input card contains a value for DELT less than or equal to zero, all succeeding event cards are scheduled in the (START, STØP) interval defined by that card until a new card with DELT less than or equal to zero is encountered.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
IGAIN	I	C	Integer flag controlling filtering algorithm IGAIN = 4 means read gain from gain file.
GAINCR	I	C	Logical flag controlling gain file creation. = .TRUE., create gain file. = .FALSE., do not create gain file
MPFREQ	I/O	C	Array of measurement print control flags.
MCØDE	O	C	Array of measurement and propagation event codes.
NSCHED	I/O	C	Input as number of scheduling cards to be read. Output as number of entries in SCHEDM MCØDE arrays to be operated on for scheduling current run.
SCHEDM	O	C	Array defining scheduling of events found in MCØDE. Each MCØDE (I) will be scheduled starting at SCHEDM (1, I), stopping at SCHEDM (2, I), in increments of SCHEDM (3, I).

Variable	Input/ Output	Argument/ Common	Definition
TCURR	I	C	Trajectory start time, lower bound for measurement scheduling.
TFINAL	I	C	Trajectory stop time, upper bound for measurement scheduling.

Local Variables:

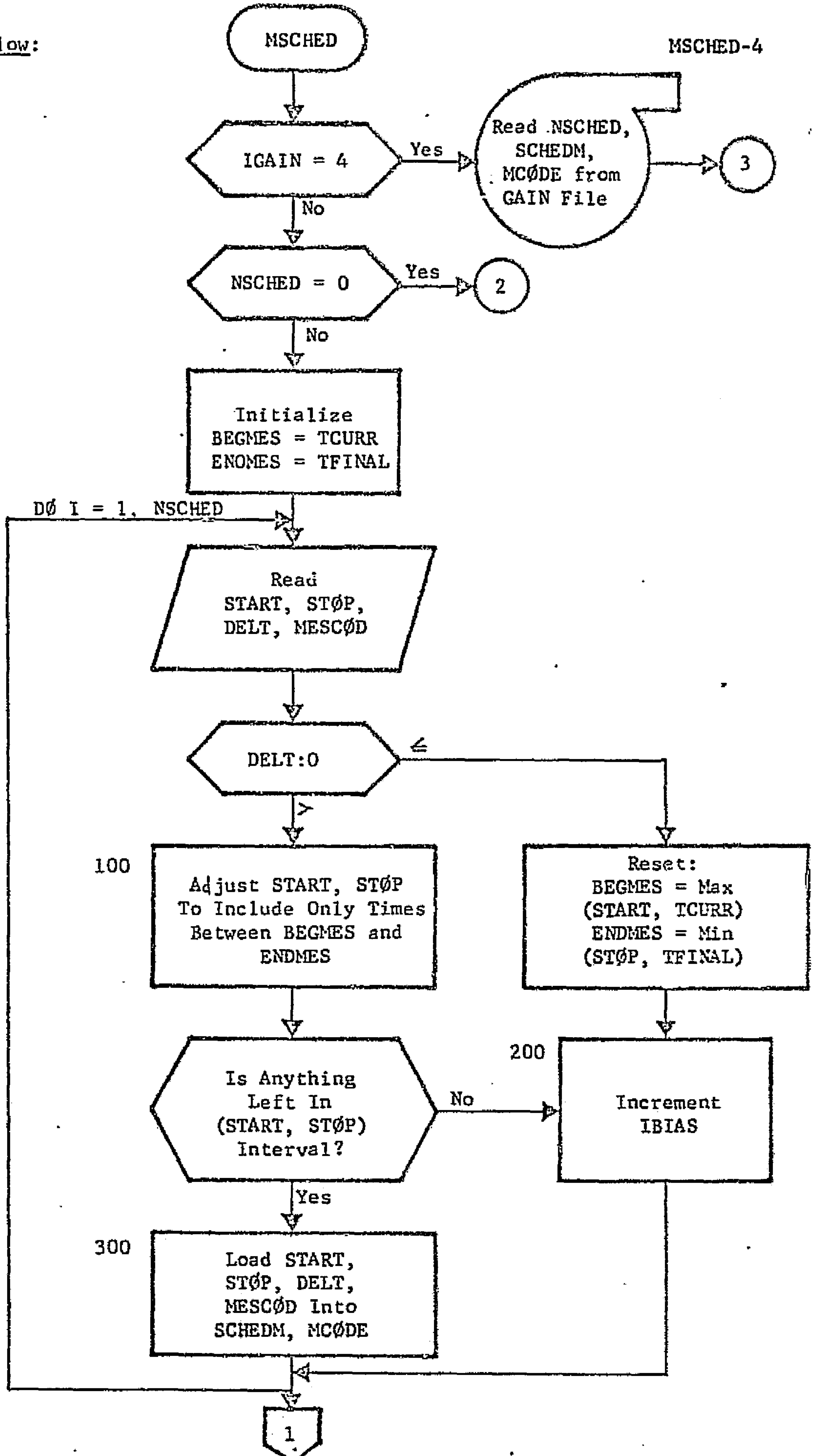
Variable	Definition
BEGMES	Beginning of allowable event scheduling interval, initially set to TCURN.
DELT	Scheduled time interval between measurements.
ENDMES	End of allowable event scheduling interval, initially set to TFINAL.
IBIAS	Running counter of number of schedule cards read but not loaded into SCHEDM and MCØDE arrays.
MESCØD	Measurement code read from input card.
START	Beginning of scheduling interval for measurement type MESCØD.
STØP	End of scheduling interval for measurement type MESCØD.

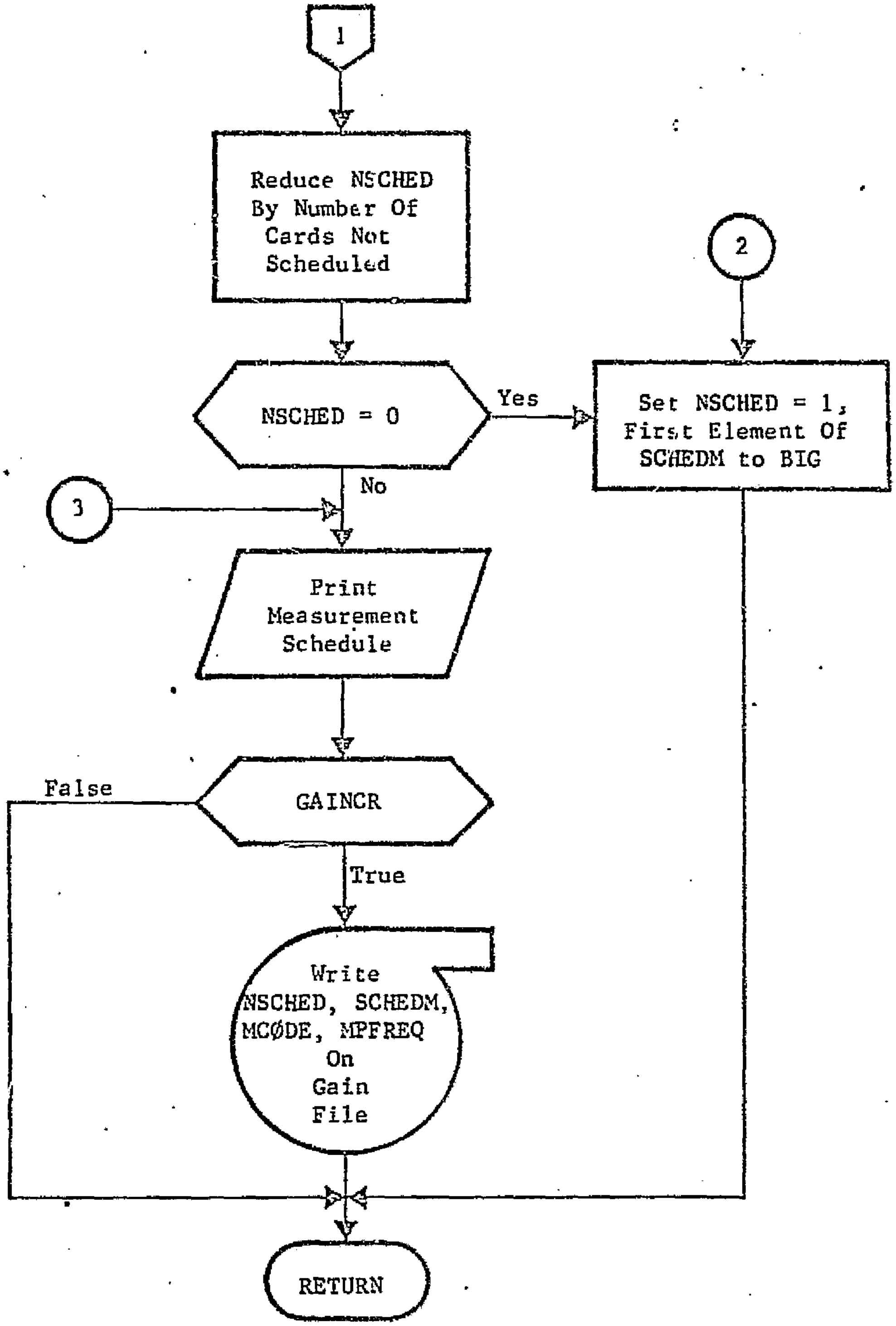
Subroutines Called: None

Calling Subroutines: ØUTPTG

Common Blocks: CØNST, SCHEDI, SCHEDR, MEASI, LØGIC, WØRK

Logic Flow:


MSCHED
IGAIN = 4
Yes
Read NSCHED, SCHEDM, MCØDE from GAIN File
3
No
NSCHED = 0
Yes
2
No
Initialize BEGMES = TCURR ENDMES = TFINAL
DØ I = 1, NSCHED
Read START, STØP, DELT, MESCØD
DELT:0
≤
>
100
Adjust START, STØP To Include Only Times Between BEGMES and ENDMES
Reset: BEGMES = Max (START, TCURR) ENDMES = Min (STØP, TFINAL)
200
Is Anything Left In (START, STØP) Interval?
No
Increment IBIAS
Yes
300
Load START, STØP, DELT, MESCØD Into SCHEDM, MCØDE
1


1
Reduce NSCHED By Number Of Cards Not Scheduled
2
NSCHED = 0
Yes
Set NSCHED = 1, First Element Of SCHEDM to BIG
No
3
Print Measurement Schedule
False
GAINCR
True
Write NSCHED, SCHEDM, MCØDE, MPFREQ On Gain File
RETURN

3.3.25 Subroutine: NMLIST

Purpose: Read $GØDSEP namelist

Remarks: All knowledge and control covariance matrix partitions are provided as arguments to NMLIST in order to minimize the number of modifications necessary in the event maximum dimensions of any sub-block are changed. Dimensions of these arrays in NMLIST must correspond to those specified for MAXDIM array in subroutine DEFALT (Sec. 3.3.8)

If GAIN file is being created, NMLIST writes all variables in namelist $GØDSEP to GAIN file (TAPE 4) in binary format. Similarly, if GAIN file is being read, NMLIST reads default values for namelist $GØDSEP in binary format from GAIN file (TAPE 4) and then reads normal namelist $GØDSEP from input to modify defaulted values as desired.

Input/Output: See GØDSEP Input, Volume II, User's Manual Sec. 2.3

Local Variables: None

Subroutines Called: JØBTLE, BØMB

Calling Subroutines: INPUTG

Common Blocks: DATAGI, DATAGR, DIMENS, GUIDE, LABEL, LØGIC, MEASI, MEASR, PRØPI, PRØPR, SCHEDI, SCHEDR, TRAJ2

Logic Flow: None

3.3.26 Subroutine: ØBSERV (HMAT)

Purpose: To compute observation matrix

Method: See Volume I, Analytical Manual, Sec. 6.3

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
HMAT	Ø	A	Observation matrix
AZMTH2	Ø	C	Azimuth angle from station ISTA2
AZMUTH	Ø	C	Azimuth angle from station ISTA1
DXDKST	I	C	Keplerian to cartesian transformation for ephemeris elements.
ECEQ	I	C	Rotation matrix from equatorial to ecliptic coordinates.
ELEV	Ø	C	Elevation angle from station ISTA1
ELEV2	Ø	C	Elevation angle from station ISTA2
GHZERØ	I	C	Greenwich hour angle at launch
IAUGST	I	C	Location in IAUG array of station location flags.
IBAZEL	I	C	Location in IAUG array of azimuth-elevation angle measurement bias flags.
IBDIAM	I	C	Location in IAUG array of apparent planet diameter measurement bias flag.
IBSTAR	I	C	Location in IAUG array of star-planet angle measurement bias flags.
IB2WAY	I	C	Location in IAUG array of 2-way range and range-rate measurement bias flags.
IB3WAY	I	C	Location in IAUG array of 3-way range and range-rate measurement bias flags.

Variable	Input/ Output	Argument/ Common	Definition
IPATYP	Ø	C	General data type decoded from MESEVN. =1, range-rate measurement =2, range measurement =3, azimuth-elevation anble measurement =4, on-board optics, star-planet angle =5, on-board optics, apparent planet diameter
IEPHEM	I	C	Ephemeris body coordinate system flag =0, non-stationary cartesian =1, stationary cartesian =2, stationary Keplerian
ISTA1	Ø	C	For IDATYP = 1,2,3 ISTA1 = station number of first station. For IDATYP=4 Number of first star. For IDATYP=5 ignored.
ISTA2	Ø	C	For IDATYP=1,2,3 ISTA2 = station number of second station (if data type requires) For IDATYP=4 number of second star. For IDATYP=5 ignored.
ISTA3	Ø	C	Ignored if IDATYP=3,4,5 If IDATYP=1,2: =0, 2-way data from station ISTA1 =1, 3-way data from stations ISTA1 and ISTA2 =2, simultaneous 2-way/3-way data from station ISTA1 and ISTA2 =3, differenced 2-way/3-way data from stations ISTA1 and ISTA2.
LIST	I	C	List of augmented parameter numbers.
LISTPH	I	C	List of ephemeris element parameter numbers.
MAXSTA	I	C	Maximum station number for which station location errors and 2-way or 3-way biases are allowed.
MESEVN	I	C	Measurement code of current data type.

Variable	Input/ Output	Argument/ Common	Definition
NAUG	I	C	Length of augmented state vector.
NB	I	C	Array of bodies used in trajectory integration.
NBØD	I	C	Number of bodies used in trajectory integration.
NEP	I	C	Number of ephemeris body.
NEPHEL	I	C	Number of ephemeris elements augmented to state.
NR	Ø	C	Length of current measurement vector.
ØMEGAG	I	C	Earth sidereal rotation rate.
PRADIS	I	C	Array of planetary radii
RAD	I	C	Conversion constant, degrees/ radian
RANGE	Ø	C	Range from station ISTA1 to S/C or range from S/C to ephemeris body.
RANGE2	Ø	C	Range from station ISTA2 to S/C
SCDEC	Ø	C	S/C geocentric equatorial declination.
SCGLØN	Ø	C	S/C geocentric longitude.
STALØC	I	C	Array of station location cylindrical coordinates.
STARDC	I	C	Array of star direction cosines.
STPANG	Ø	C	Array of star planet angles.
TCURR	I	C	Current trajectory time.
TM	I	C	Conversion constant, seconds/day.
UP	I	C	Position array of bodies used in trajectory integration.

Variable	Input/ Output	Argument/ Common	Definition
UREL	I	C	Relative position array of S/C to bodies for trajectory integration.
VP	I	C	Velocity array of bodies used in trajectory integration.
VREL	I	C	Relative velocity array of S/C to bodies for trajectory integration.

Local Variables: For all variables and equations, see Volume 1, Analytical Manual, Section 6.

Variable	Definition
CACB	COS (azimuth) x COS (elevation)
CALPHA	COS (azimuth)
CBETTA	COS (elevation)
CGAMMA	COS (star-planet angle)
DABDX	$\partial(\alpha,\beta)/\partial x$
DABDXS	$\partial(\alpha,\beta)/\partial \underline{x}_S$
DELR	Vector position difference between stations ISTA1 and ISTA2.
DELRHØ	$\Delta\rho$
DIFF23	Logical flag =T, differenced 2-way/3-way data =F, not differenced 2-way/3-way data
DØPLER	Logical flag =T, range-rate measurement =F, not range-rate measurement
GECSTA	Geocentric ecliptic coordinates of ISTA1
GECST2	Geocentric ecliptic coordinates of ISTA2
GECV	S/C geocentric ecliptic coordinates
GEQSTA	Geocentric equatorial coordinates of ISTA1

Variable	Definition
GEQV	S/C Geocentric equatorial coordinates
HECE	Heliocentric ecliptic coordinates of Earth.
HECP	Heliocentric ecliptic coordinates of ephemeris body.
HECV	S/C Heliocentric ecliptic coordinates.
HV	Observation partials for ISTA1 station location parameters.
HV2	Observation partials for ISTA2 station location parameters.
HX	2-way observation partials for S/C state from ISTA1.
HX2	2-way observation partials for S/C state from ISTA2.
ISTA	Number of station or star for which partials are currently being computed.
NTEMP	When multi-station data is used, information for ISTA2 is computed first in locations HX,HV,RHØHAT, and GECSTA. NTEMP is number of words which must be copied from HX, etc. into HX2, etc.
PARKEP	Observation sensitivities to ephemeris body elements.
PECCYL	Partial of instantaneous station geocentric ecliptic to cylindrical coordinates.
PEQCYL	Partial of instantaneous station geocentric equatorial to cylindrical coordinates.
RHO	Range vector from station ISTA to S/C or from S/C to ephemeris body.
RHØDØT	Relative velocity vector from station ISTA to S/C.

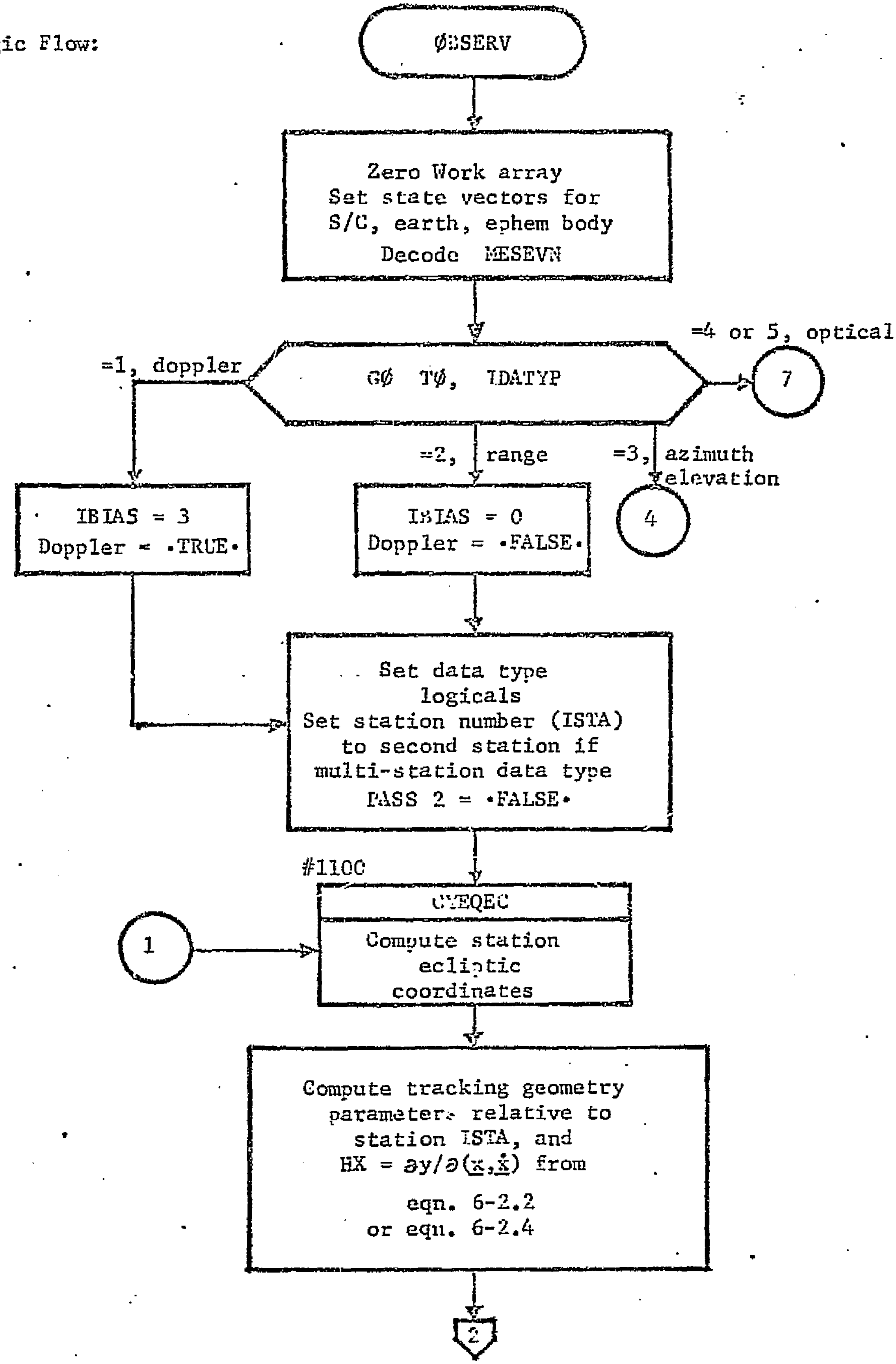
Variable	Definition
RHØHAT	Unit vector in RHØ direction from ISTA1
RHØHT2	Unit vector in RHØ direction from ISTA2
SALPHA	SIN α
SBETA	SIN β
SGAMMA	SIN γ
SGNCØS	Signum (COS α)
SIML23	Logical flag =T, simultaneous 2-way/3-way data =F, not simultaneous 2-way/3-way data
SINE	Sin (apparent planet diameter angle)
TATB	tan α tan β
THRWAY	Logical flag =T, 3-way data only =F, not 3-way data only
TWØWAY	Logical flag =T, 2-way data only =F, not 2-way data only
WHAT	$\hat{w}$
XSHAT	$\hat{X}_s$

Subroutines Called: ZERØM, CYEQEC, VECMAG, UNITV, UDØTV, ASIN, LØCLST, PARSTA, MMAB, NEGMAT, MMATB, ATAN2, CØPY, ADD, MUNPAK, SUB, UXV, SQRT, MMABT, ACØS, LØDCØL

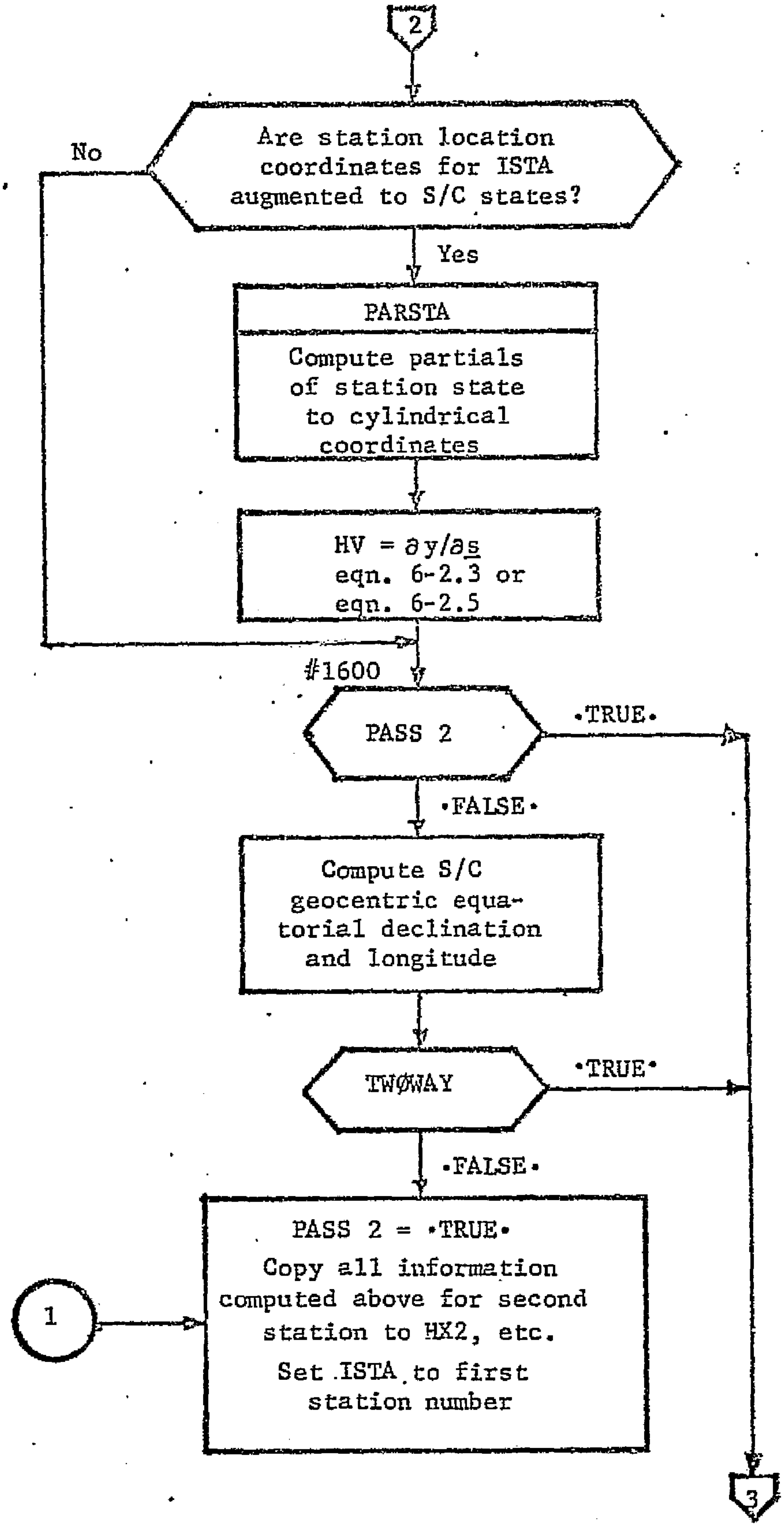
Calling Subroutines: MEAS

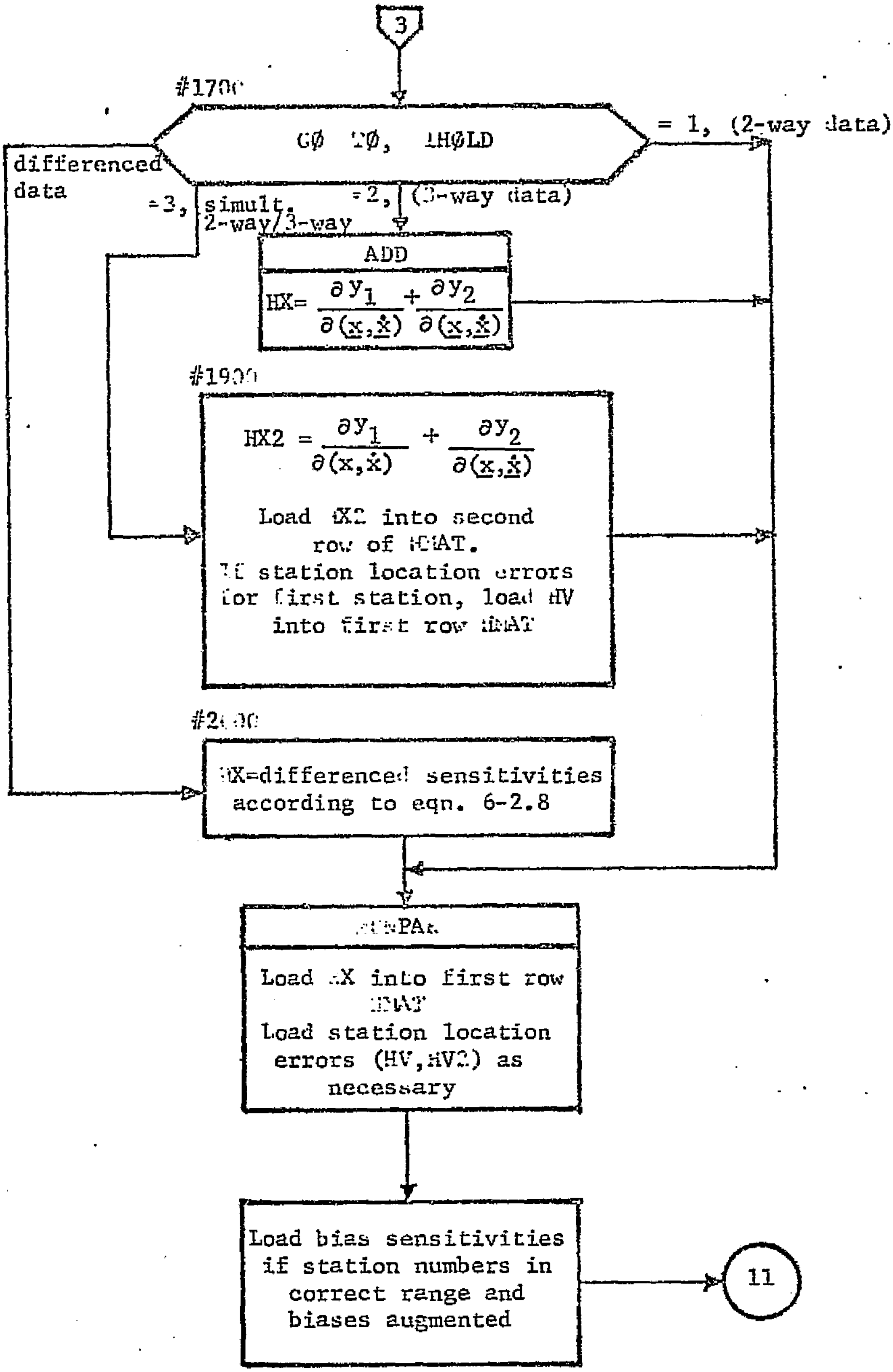
Common Blocks: WØRK, (BLANK), CØNST, DIMENS, EPHEM, KEPCØN, MEASI, MEASR, SCHEDI, SCHEDR, TRAJ1, TRAJ2

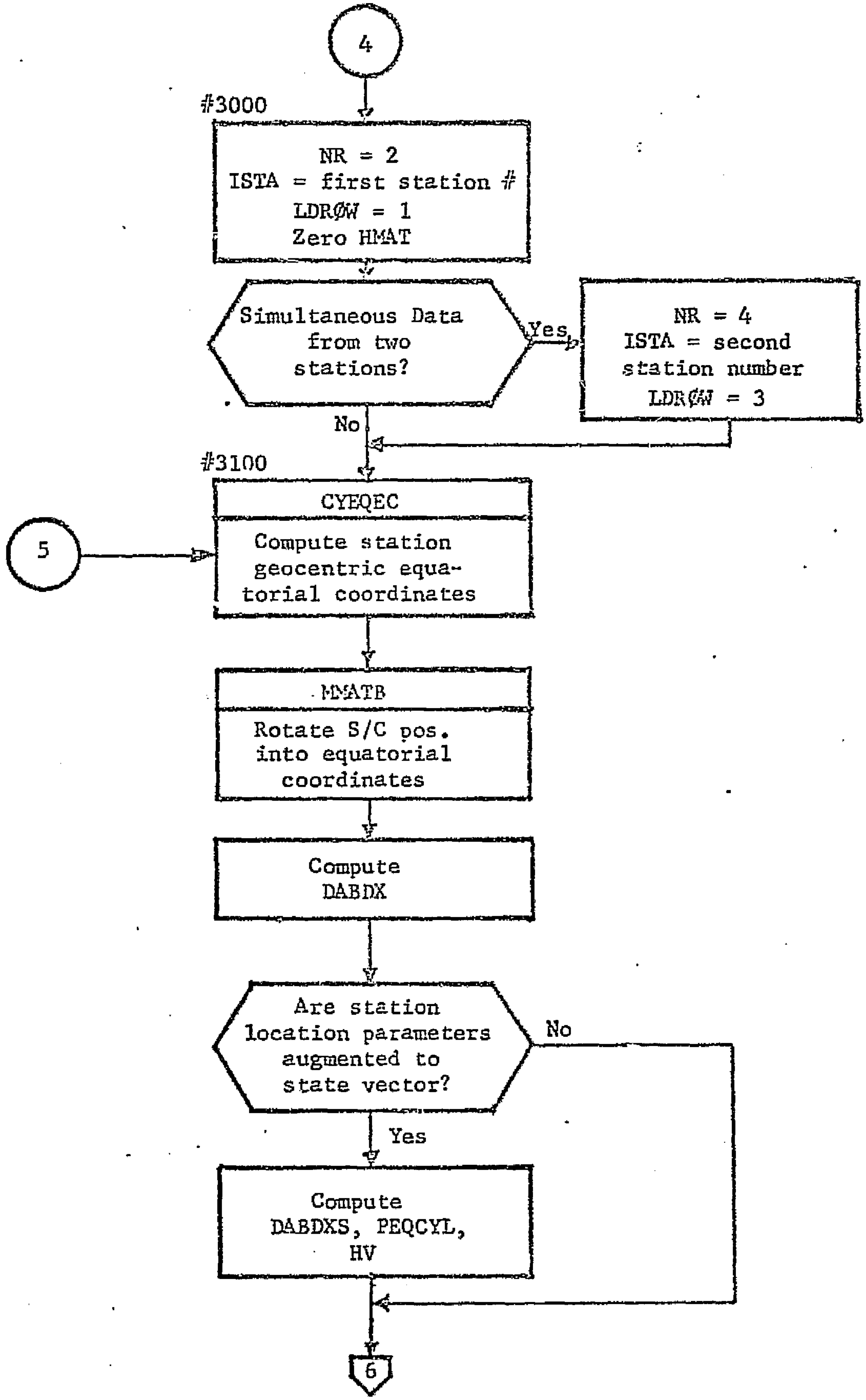
Logic Flow:



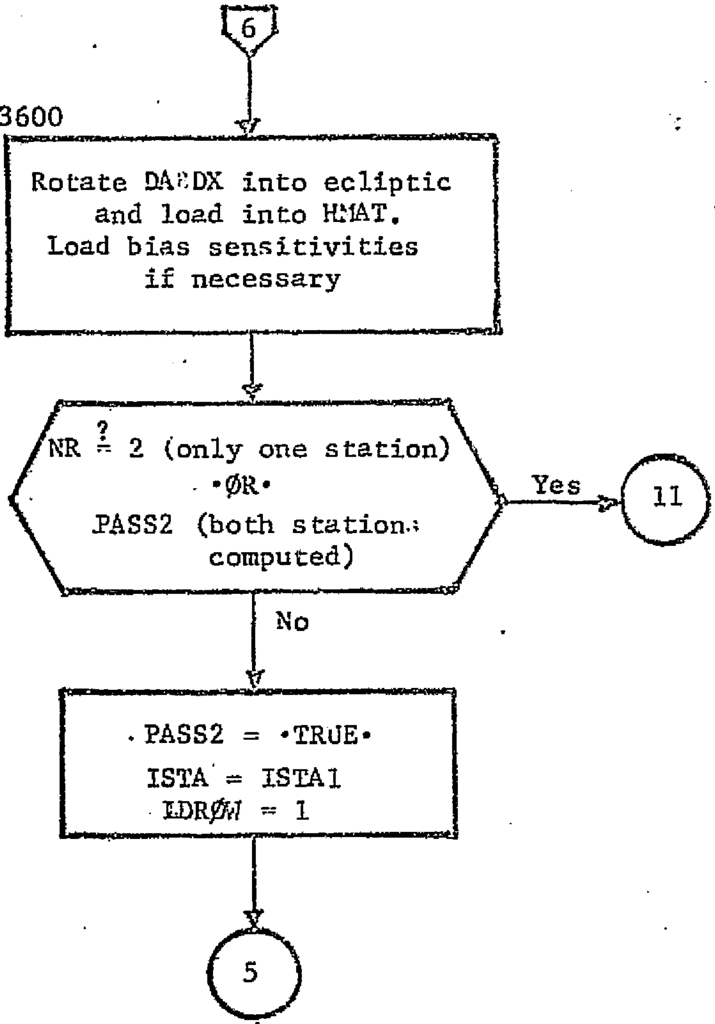
ØBSERV
Zero Work array
Set state vectors for
S/C, earth, ephem body
Decode MESEVN
=1, doppler
GØ TØ, IDATYP
=4 or 5, optical
7
=2, range
=3, azimuth
elevation
4
IBIAS = 3
Doppler = .TRUE.
IBIAS = 0
Doppler = .FALSE.
Set data type
logicals
Set station number (ISTA)
to second station if
multi-station data type
PASS 2 = .FALSE.
#1100
CYEQEC
Compute station
ecliptic
coordinates
1
Compute tracking geometry
parameters relative to
station ISTA, and
HX = ∂y/∂(x,ẋ) from
eqn. 6-2.2
or eqn. 6-2.4
2


2
No
Are station location coordinates for ISTA augmented to S/C states?
Yes
PARSTA
Compute partials of station state to cylindrical coordinates
HV = ∂y/∂s
eqn. 6-2.3 or
eqn. 6-2.5
#1600
PASS 2
•TRUE•
•FALSE•
Compute S/C geocentric equatorial declination and longitude
TWØWAY
•TRUE•
•FALSE•
PASS 2 = •TRUE•
Copy all information computed above for second station to HX2, etc.
Set ISTA to first station number
1
3

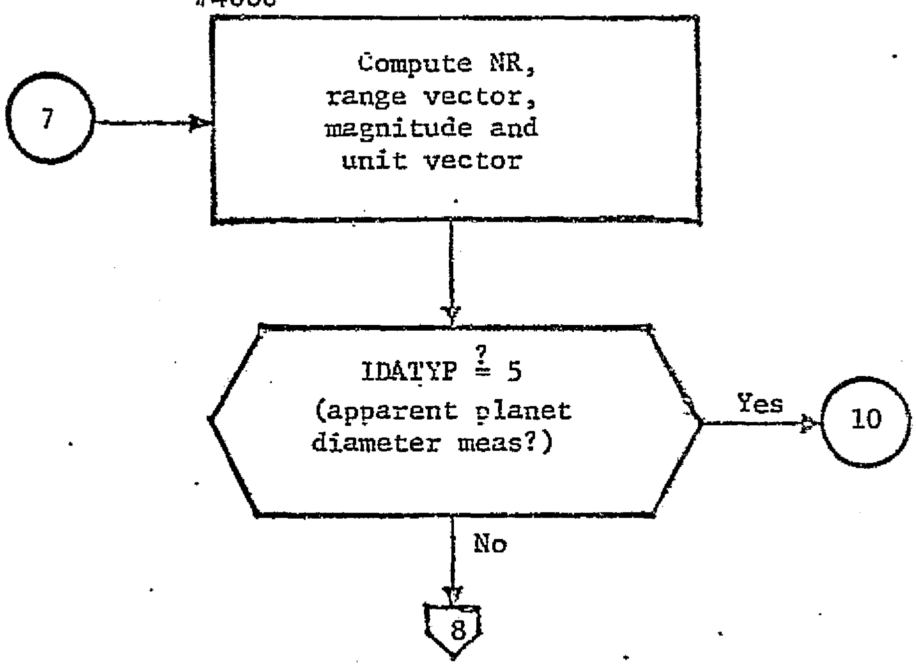

3
#1700
GØ TØ, 1HØLD
= 1, (2-way data)
differenced data
=3, simult. 2-way/3-way
=2, (3-way data)
ADD
$HX = \frac{\partial y_1}{\partial (\underline{x}, \dot{\underline{x}})} + \frac{\partial y_2}{\partial (\underline{x}, \dot{\underline{x}})}$
#1900
$HX2 = \frac{\partial y_1}{\partial (x, \dot{x})} + \frac{\partial y_2}{\partial (\underline{x}, \dot{\underline{x}})}$
Load HX2 into second row of HMAT.
If station location errors for first station, load HV into first row HMAT
#2000
HX=differenced sensitivities according to eqn. 6-2.8
Load HX into first row HMAT
Load station location errors (HV, HV2) as necessary
Load bias sensitivities if station numbers in correct range and biases augmented
11

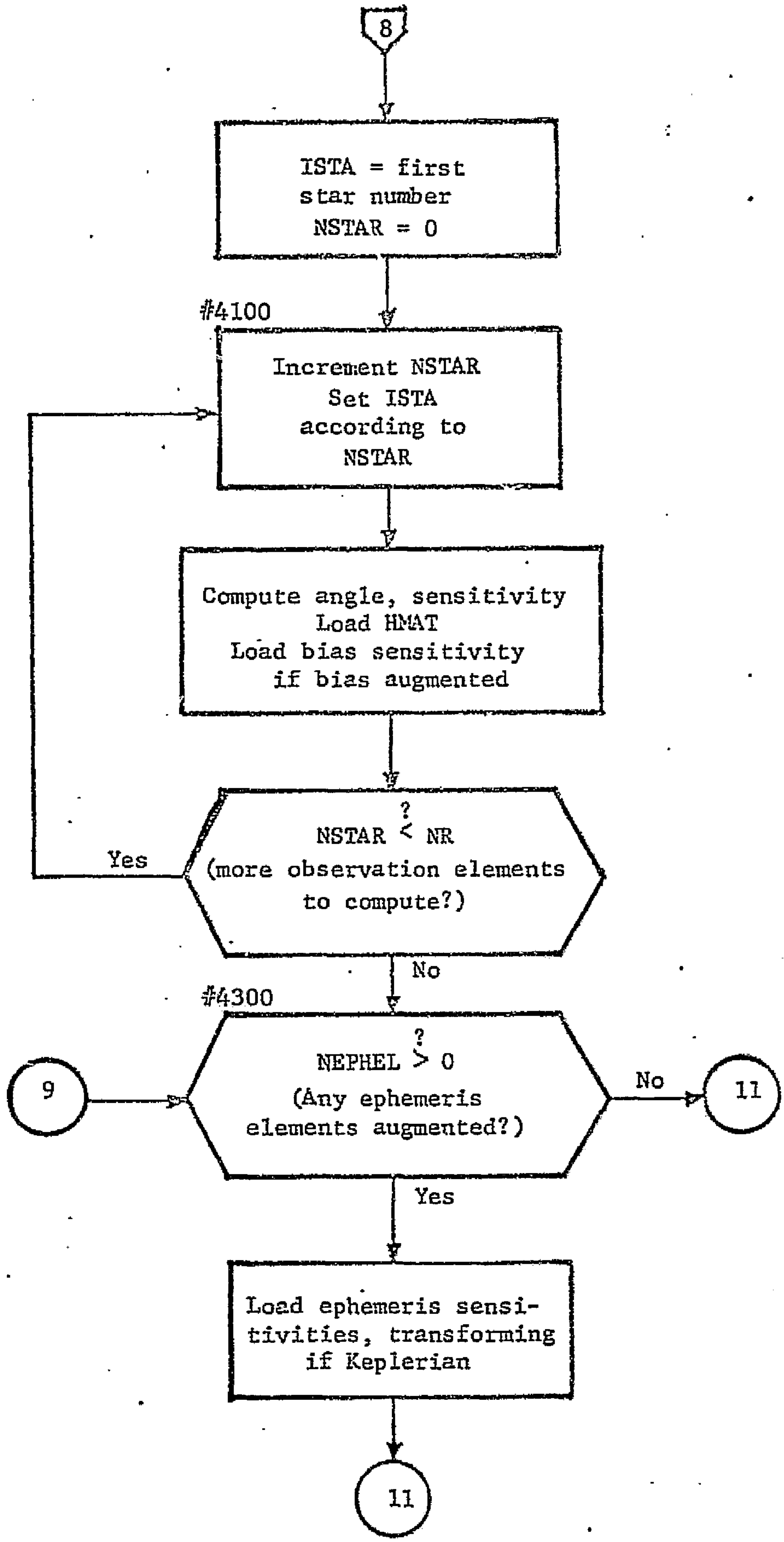

4
#3000
NR = 2
ISTA = first station #
LDRØW = 1
Zero HMAT
Simultaneous Data from two stations?
Yes
NR = 4
ISTA = second station number
LDRØW = 3
No
#3100
CYEQEC
Compute station geocentric equatorial coordinates
5
MMATB
Rotate S/C pos. into equatorial coordinates
Compute DABDX
Are station location parameters augmented to state vector?
No
Yes
Compute DABDXS, PEQCYL, HV
6

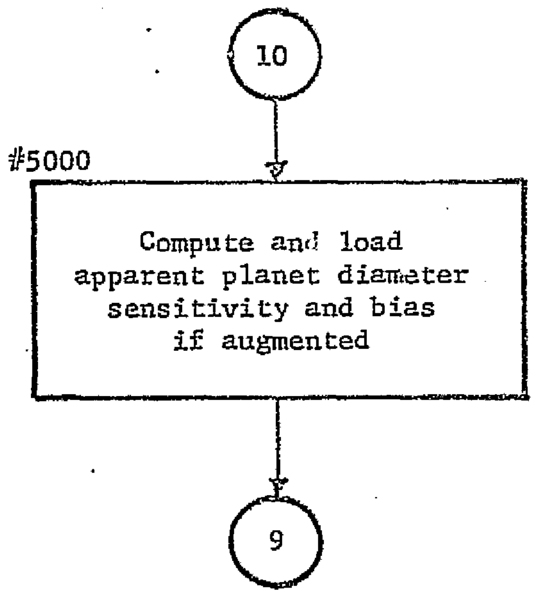
#3600

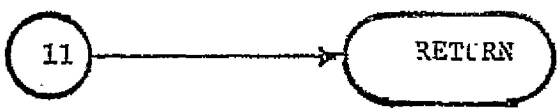

6
Rotate DAEDX into ecliptic and load into HMAT. Load bias sensitivities if necessary
NR ≟ 2 (only one station) •ØR• PASS2 (both stations computed)
Yes
11
No
PASS2 = •TRUE•
ISTA = ISTA1
LDRØW = 1
5

#4000


7
Compute NR, range vector, magnitude and unit vector
IDATYP ≟ 5 (apparent planet diameter meas?)
Yes
10
No
8


8
ISTA = first
star number
NSTAR = 0
#4100
Increment NSTAR
Set ISTA
according to
NSTAR
Compute angle, sensitivity
Load HMAT
Load bias sensitivity
if bias augmented
NSTAR ? < NR
(more observation elements
to compute?)
Yes
No
#4300
9
NEPHEL ? > 0
(Any ephemeris
elements augmented?)
No
11
Yes
Load ephemeris sensi-
tivities, transforming
if Keplerian
11


10
#5000
Compute and load
apparent planet diameter
sensitivity and bias
if augmented
9


11
RETURN

3.3.27 Subroutine: ØUTPTG

Purpose: Print out for user information of options selected and initial values. Conversion of input to internal units as necessary.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
BIG	I	C	Large constant, 1.E20.
CØNRD	I	C	Logical flag. = T, control uncertainties read in. = F, control uncertainties not read in.
CØRLØN	I	C	Station longitude correlation coefficient.
DOPCNT	I	C	Average number of range-rate measurements taken per day during tracking arc.
DYNØIS	I	C	Logical flag. = T, compute effective process noise. = F, do not compute effective process noise.
EPSIG	I	C	Array of process noise standard deviations.
EPTAU	I	C	Array of process noise correlation times.
EPVAR	I	C	Array of process noise variances.
GAINCR	I	C	Logical flag. = T, create GAIN file. = F, do not create GAIN file.

Variable	Input/ Output	Argument/ Common	Definition
GENCOV	I	C	Logical flag. = T, generalized covariance analysis on current run. = F, no generalized covariance analysis on current run.
GTAU1	Ø	C	Array of negative inverse primary process noise correlation times for TRAJ (Section [illegible]) Operative only if PDØT = .TRUE.
GTAU2	Ø	C	Array of negative inverse secondary process noise correlation times for TRAJ (Section [illegible]) Operative only if PDØT = .TRUE.
IAUGST	I	C	Location in IAUG array of station location parameters.
IGAIN	I	C	Gain matrix algorithm flag.
ISTMF	I	C	STM file usage flag.
LIST	I	C	Array of augmented parameter numbers.
LPDØT	Ø	C	Array of dynamic parameters to TRAJ (Section [illegible]) Operative only if PDØT = .TRUE.
MCØUNT	Ø	C	Measurement counter.
MPFREQ	I/O	C	Measurement print frequency control array.
NAUG	I	C	Length of augmented state vector.
NCNTE	Ø	C	Eigenvector event counter.

Variable	Input/ Output	Argument/ Common	Definition
NENTG	Ø	C	Guidance event counter.
NCNTP	Ø	C	Prediction event counter.
NCNTT	Ø	C	Thrust event counter.
NEIGEN	I/Ø	C	Total number of eigenvector events to be scheduled.
NGUID	I/Ø	C	Total number of guidance events to be scheduled.
NPRED	I/Ø	C	Total number of prediction events to be scheduled.
NTHRST	I/Ø	C	Total number of thrust events to be scheduled.
NST	I	C	Number of tracking stations defined.
P	I	C	Location in blank common of knowledge covariance.
PDOT	I	C	Logical flag. = T, covariance propagation by integration of variational equations. = F, covariance propagation by state transition matrices.
PG	I	C	Location in blank common of control covariance.
PGLAB	I	C	Array of control covariance sub-block Hollerith labels.
PLAB	I	C	Array of knowledge covariance sub-block to Hollerith labels.
PRNCØV	I	C	Logical array controlling covariance sub-blocks printed.
PRØPG	Ø	C	Logical flag.

Variable	Input/ Output	Argument/ Common	Definition
			= T, propagate control covariance simultaneously with knowledge. = F, do not propagate control covariance simultaneously with knowledge.
QNØISE	Ø	C	Array of process noise variances provided to TRAJ (Section 3.5) when PDØT = .TRUE.
RAD	I	C	Conversion constant, degrees/radian.
SCHFTL	I	C	Logical flag. = T, mesh failure on reading STM file is fatal. = F, mesh failure on reading STM file is not fatal.
SIGLØN	I	C	Standard deviation in station longitude.
SIGMES	I	C	Array of measurement white noise standard deviations.
SIGRS	I	C	Standard deviation in station spin radius.
SIGZ	I	C	Standard deviation in station z-height.
STALØC	I	C	Array of tracking station cylindrical coordinates.
TCURR	I	C	Current (and initial) trajectory time.
TDUR	I	C	Trajectory final time (seconds) for TRAJ (Section 3.5)
TEIGEN	I	C	Array of eigenvector event times.

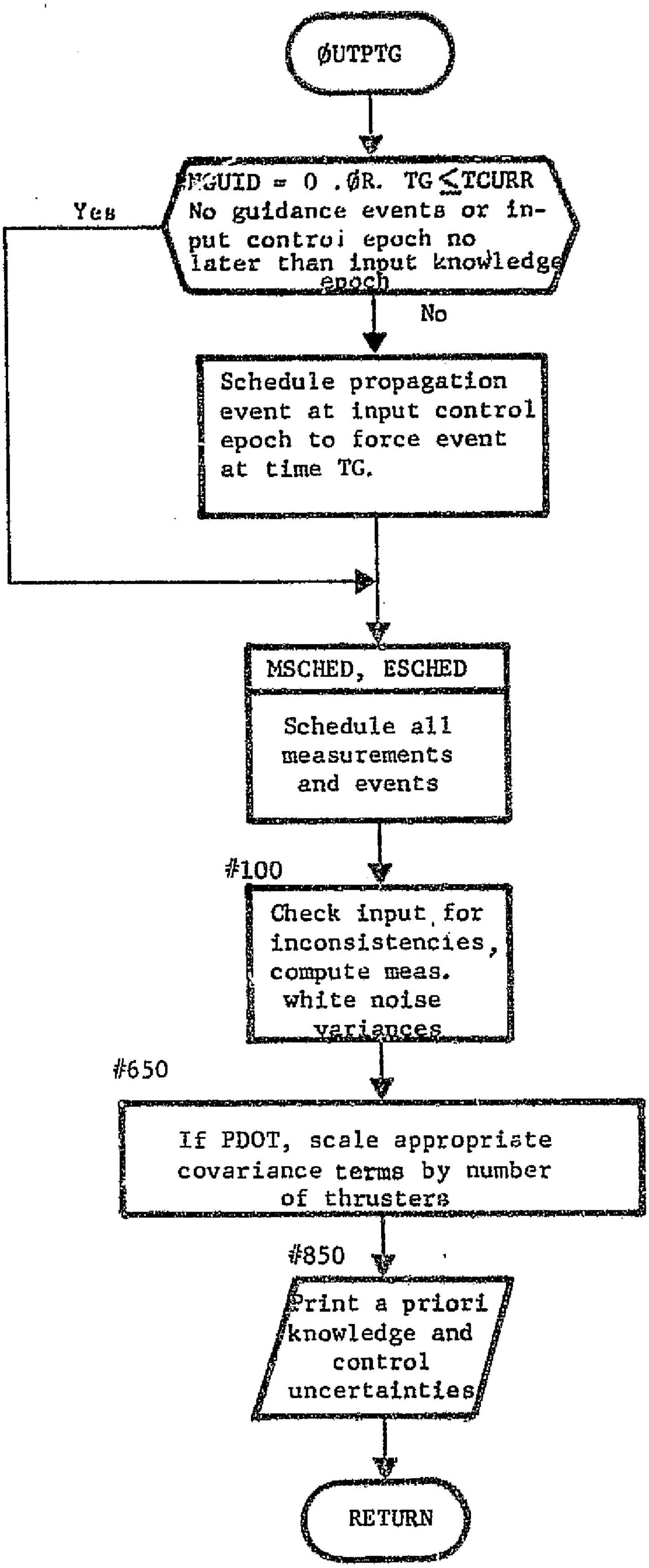
Variable	Input/ Output	Argument/ Common	Definition
TFINAL	I	C	Error analysis final time.
TG	I	C	Epoch for input control uncertainties if CØNRD = .TRUE.
TGUID	I	C	Array of guidance event times.
TM	I	C	Conversion constant, seconds/day.
TØLBAK	I	C	Backward tolerance on STM file mesh.
TØLFØR	I	C	Forward tolerance on STM file mesh.
TPRED	I	C	Array of prediction event times.
TTHRST	I	C	Array of thrust event times.
VARMES	Ø	C	Array of measurement white noise variances.

Local Variables: No..

Subroutines Called: MSCHED, ESCHED, SCHED, BØMB, ATAN, ZERØM, CØRREL, SDVAR, CØPY

Calling Subroutines: DATAG

Common Blocks: WØRK, (BLANK), CØNST, DATGI, DATGR, DIMENS, LABEL, LØCATE, LØGIC, MEASI, MEASR, PRØPI, PRØPR, SCHEDI, SCHEDR, TIME, TRAJ1, TRAJ2


ØUTPTG
NGUID = 0 .ØR. TG ≤ TCURR
No guidance events or input control epoch no later than input knowledge epoch
Yes
No
Schedule propagation event at input control epoch to force event at time TG.
MSCHED, ESCHED
Schedule all measurements and events
#100
Check input for inconsistencies, compute meas. white noise variances
#650
If PDOT, scale appropriate covariance terms by number of thrusters
#850
Print a priori knowledge and control uncertainties
RETURN

3.3.28 Subroutine: PARKEP (X, V, GMU, DXDK)

Purpose: To compute Keplerian to cartesian transformation.

Method: Central differencing.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
X	I	A	Cartesian position of body.
V	I	A	Cartesian velocity of body.
GMU	I	A	Gravitational constant of central body.
DXDK	Ø	A	Output Keplerian to Cartesian transformation.

Local Variables:

Variable	Definition
XKEP	Body Keplerian elements.
XMINUS	Body cartesian state using negative perturbation.
XPLUS	Body cartesian state using positive perturbation.
XPERT	Array of cartesian perturbation levels.

Subroutines Called: CØNIC, CARTES

Calling Subroutines: SETGUI, STMRDR, CØVP

Common Blocks: WØRK

Logic Flow: None

3.3.29 Subroutine: PARSTA (GEQSTA, RSPIN, ECEQ, PECCYL)

Purpose: To compute the partial derivative of station instantaneous geocentric ecliptic cartesian state wrt station equatorial cylindrical coordinates.

Method: See Volume 1, Analytical Manual, Section 6.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
GEQSTA	I	A	Instantaneous geocentric equatorial cartesian state of station.
RSPIN	I	A	Station spin radius.
ECEQ	I	A	Rotation matrix from equatorial to ecliptic cartesian system.
PECCYL	Ø	A	Partial derivative of instantaneous ecliptic state of state wrt cylindrical coordinates.

Local Variables:

Variable	Definition
CØSEPS, SINEPS	COS and SIN of Earth obliquity to ecliptic.
CØSPHI, SINPHI	COS and SIN of instantaneous station equatorial longitude.
CPØMEG, SPØMEG	COS and SIN of Earth inertial rotation rate.

Subroutines Called: None

Calling Subroutines: ØBSERV

Common Blocks: None

Logic Flow: None

3.3.30 Logical Function: PCNTRL (ITYPE, ISUB)

Purpose: To control measurement print.

Method: Each general data type (e.g., 2-way range, simultaneous 2-way/3-way doppler, azimuth-elevation angles) is assigned a print frequency (MPFREQ) and a counter (MPCNTR). A test is made on the counter for the input data type defined by ITYPE, ISUB. If the MPCNTR, modulo its MPFREQ, is zero, the measurement is printed.

Remarks: Two additional features are provided. The first processed measurement of any data type whose corresponding MPFREQ element is non-zero is printed. Also, the final measurement, independent of the data type, is printed.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
ITYPE	I I	A	Basic data type, corresponds to IDATYP in common block MEASI. = 1, doppler = 2, range = 3, azimuth-elevation angle = 4, star-planet angle = 5, apparent planet diameter.
ISUB	I	A	Sub-data type for doppler and range, ignored if ITYPE > 2.

Variable	Input/ Output	Argument/ Common	Definition
			= 0, 2-way = 1, 3-way = 2, simultaneous 2-way/ 3-way = 3, differenced 2-way/ 3-way
PCNTRL	O	F*	Logical print control variable. = .TRUE., if measurement to be printed = .FALSE., if measurement not to be printed.
MPCNTR	I/O	C	Array of data type counters.
MPFREQ	I	C	Array of data type print frequencies.
TFINAL	I	C	Trajectory final time.
TMNEXT	I	C	Time of next scheduled measurement.

Local Variables:

Variable	Definition
ICØDE	Integer subscript locating data type in MPFREQ and MPCNTR.

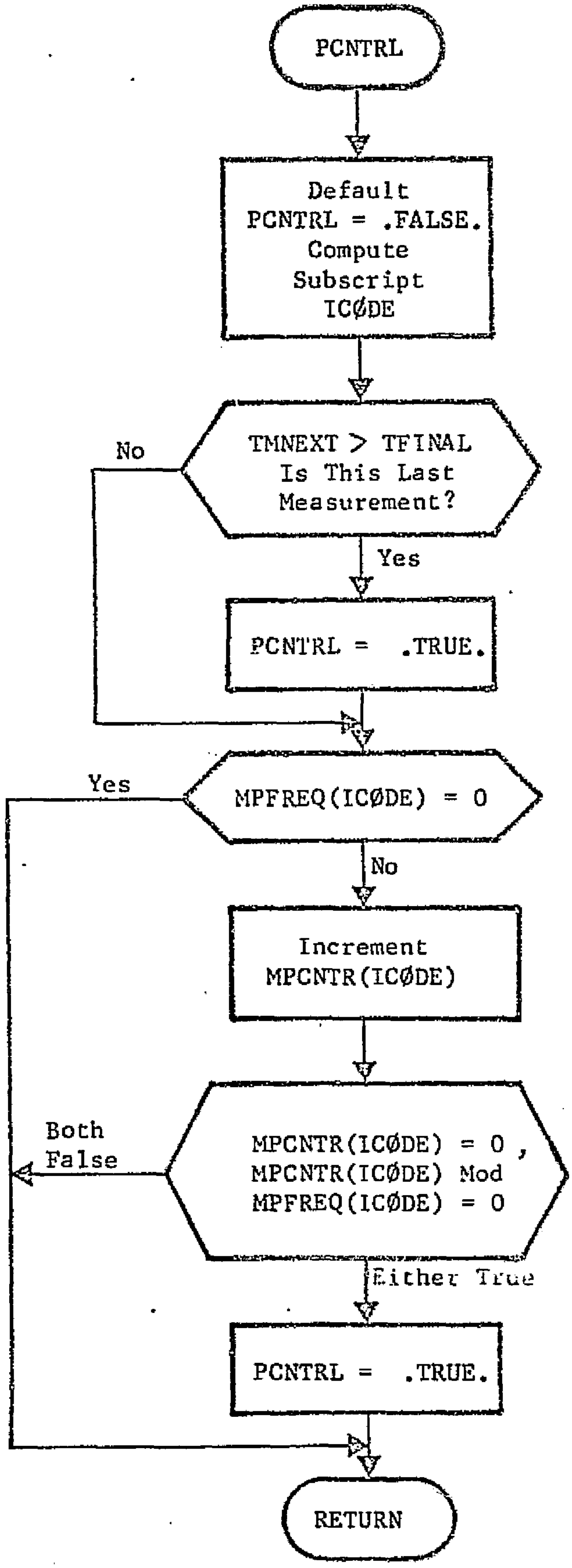
Subroutines Called: None

Calling Subroutines: MEAS

Common Blocks: SCHEDR, SCHEDI

*Function Value Output.

Logic Flow:


PCNTRL
Default
PCNTRL = .FALSE.
Compute
Subscript
ICØDE
TMNEXT > TFINAL
Is This Last
Measurement?
No
Yes
PCNTRL = .TRUE.
MPFREQ(ICØDE) = 0
Yes
No
Increment
MPCNTR(ICØDE)
MPCNTR(ICØDE) = 0 ,
MPCNTR(ICØDE) Mod
MPFREQ(ICØDE) = 0
Both
False
Either True
PCNTRL = .TRUE.
RETURN

3.3.31 Subroutine: PPAK (PBLØCK, IFØRM, PAUG)

Purpose: To load input covariances from either packed or unpacked input form to block form (See AUGCNV, Section 3.3.1).

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
PBLØCK	I	A	Array containing all input covariance information.
IFØRM	I	A	Flag indicating input form of individual sub-blocks within PBLØCK. = 1, sub-blocks are packed. = -1, sub-blocks are not packed.
PAUG	Ø	A	Output covariance in "block" form.
LØCBLK	I	C	Array locating covariance sub-blocks in "block" form (PAUG).
MAXDIM	I	C	Array of dimensions of covariance sub-blocks in PBLØCK. MAXDIM remains at input values if input sub-blocks are not packed and MAXDIM is adjusted to NDIM if sub-blocks are packed.
NDIM	I	C	Array of assumed sub-block dimensions on output.

Local Variables:

Variable	Definition
IBLØCK	Running counter locating current covariance sub-block within PBLØCK.

Variable	Definition
MAXSAV	Array saving input values of MAXDIM.

Subroutines Called: MPAK, SYMLØ, AUGCNV

Calling Subroutine: INPUTG

Common Blocks: WØRK, DATAGI, DIMENS

Logic Flow: None

3.3.32 Subroutine: PRØP (PIN, PHIMAT, NP, WLSREF, PØUT)

Purpose: To propagate an augmented covariance matrix between time points.

Method: State transition matrix with effective process noise model.

Remarks: PIN and PØUT may not share the same location. This routine also propagates the reference covariance for sequential weighted least squares (WLS) filtering.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
PIN	I	A	Input covariance to be propagated.
PHIMAT	I	A	Transition matrix over time interval.
NP	I	A	Demension of input transition matrices.
WLSREF	I	A	Logical flag controlling propagation of WLS reference covariance. = .TRUE. and IGAIN = 2, WLS reference propagated, otherwise not.
PØUT	O	A	Output covariance.
DYNØIS	I	C	Logical flag controlling addition of effective process noise. = .TRUE., add Q = .FALSE., do not add Q

Variable	Input/ Output	Argument/ Common	Definition
IGAIN	I	C	Integer flag controlling filtering algorithm = 2, use WLS ≠ 2, do not use WLS.
NSØLVE	I	C	Total number of variables solved-for (=6 + number of solve-for parameters).
PWLS	I	C	Location in blank common of WLS reference covariance.
Q	I	C	Effective dynamic noise matrix.

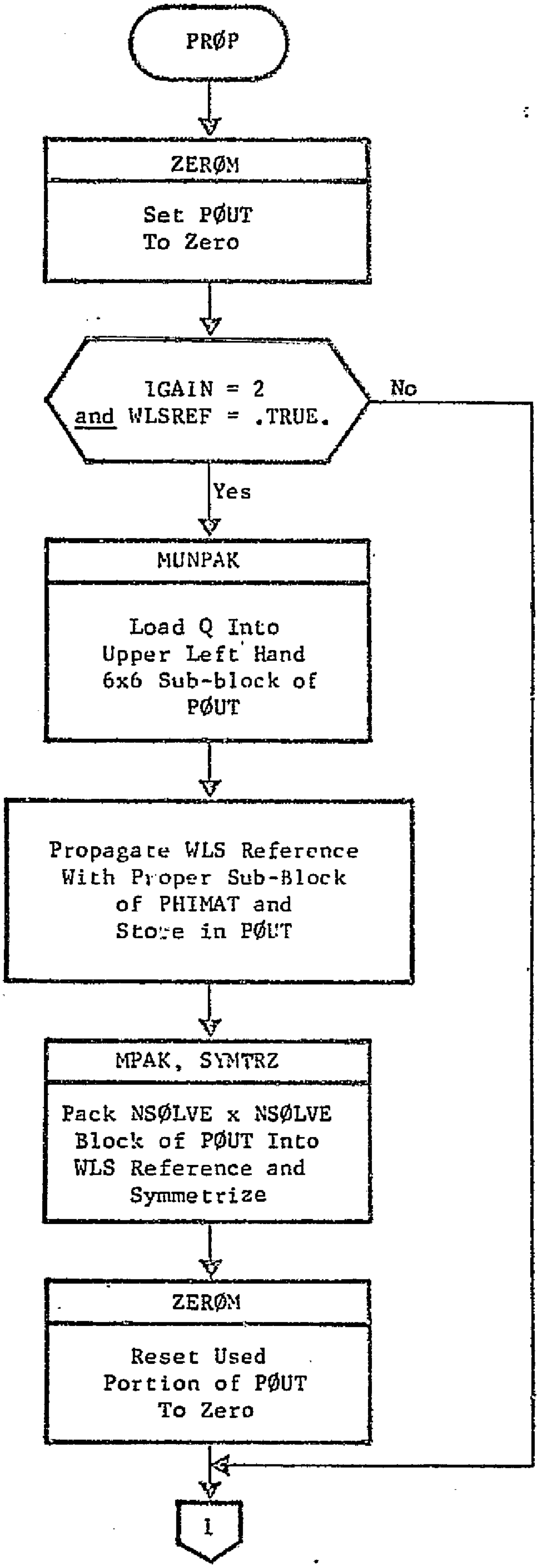
Local Variables: None

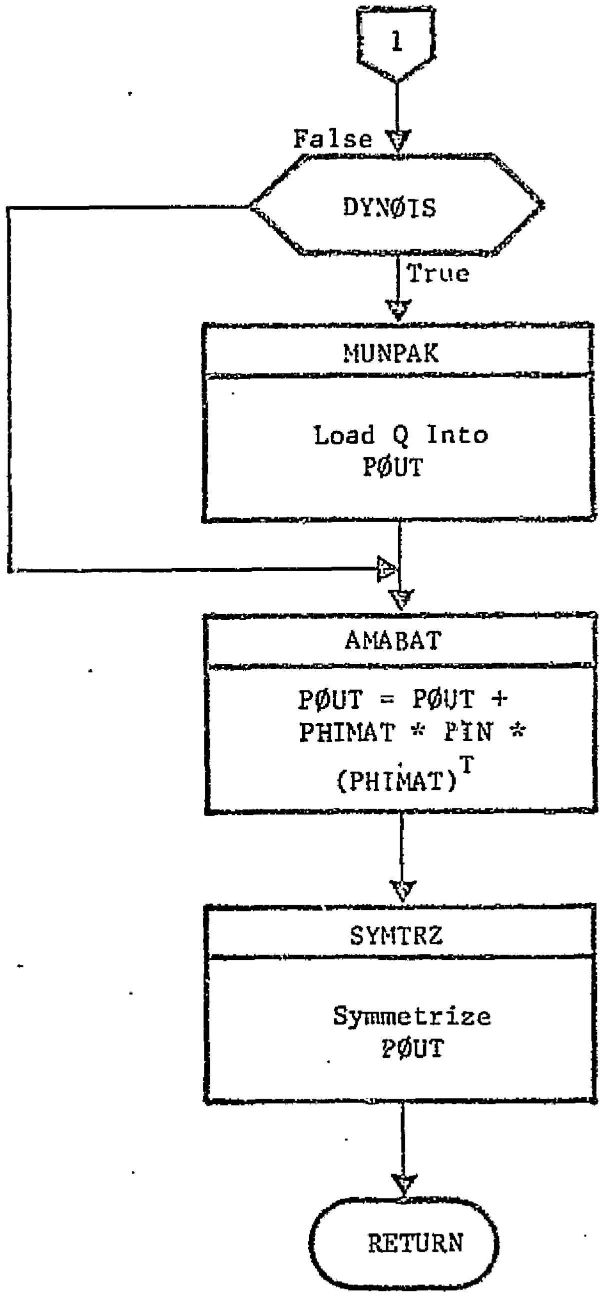
Subroutines Called: ZERØM, MUNPAK, MPAK, SYMTRZ, AMABAT

Calling Subroutines: CØVP, PRED, GUIDE

Common Blocks: (BLANK), DIMENS, LØCATE, LØGIC, MEASI, PRØPR

Logic Flow:


PRØP
ZERØM
Set PØUT To Zero
IGAIN = 2 and WLSREF = .TRUE.
No
Yes
MUNPAK
Load Q Into Upper Left Hand 6x6 Sub-block of PØUT
Propagate WLS Reference With Proper Sub-Block of PHIMAT and Store in PØUT
MPAK, SYMTRZ
Pack NSØLVE x NSØLVE Block of PØUT Into WLS Reference and Symmetrize
ZERØM
Reset Used Portion of PØUT To Zero
1


1
False
DYNØIS
True
MUNPAK
Load Q Into
PØUT
AMABAT
PØUT = PØUT +
PHIMAT * PIN *
(PHIMAT)^T
SYMTRZ
Symmetrize
PØUT
RETURN

3.3.33 Subroutine: PRPART (A, MAXRØW, NRØW, NCØL, LABEL)

Entry Points: PRCØRR, PUNCØR

Purpose: To print or punch the transpose of any sub-block or partition of a matrix with column labels for printing and a single matrix name for punching.

Remarks: This routine was designed primarily for printing partitions of covariance and transition matrices and punching covariance partitions. However, it has general applications to any matrix. PRPART and PRCØRR are functionally equivalent - the difference in output being E format by PRPART for general matrices and F format by PRCØRR for easy reading of correlation coefficients. PUNCØR punches, and is valid for general matrices. The calling sequence requires that the argument A be the first word of the partition of interest. For example, given a 9 x 9 state transition matrix, PHI, which is theoretically partitioned as

$$PHI = \begin{bmatrix} \Phi_{6x6} & \theta_{6x3} \\ 0_{3x6} & I_{3x3} \end{bmatrix}$$

to print the transpose of the Φ_{6x6} partition we would use

CALL PRPART (PHI, 9, 6, 6, LABEL1)

where LABEL1 is a 6-vector of Hollerith labels for the columns of Φ_{6x6}. Similarly to print the transpose of θ_{6x3}, we would use

CALL PRPART (PHI (1, 7), 9, 6, 3, LABEL2)

where PHI (1, 7) represents the first element of the θ_{6x3} partition, and LABEL2 as a 3-vector of Hollerith labels for the columns of θ_{6x3}. If PHI is not explicitly dimensioned 9 x 9 in the calling routine, this last call could also have been

CALL PRPART (PHI (NPHI * (7-1) + 1),
NPHI, 6, 3, LABEL2)

where the PHI subscript (NPHI * (7-1) + 1) comes from the general formula for locating element (I, J) in a matrix dimensioned (M, N):

LØC = M * (J-1) + I.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
A	I	A	First word of matrix sub-block to be printed or punched.
MAXRØW	I	A	Number of rows in complete matrix from which partition is being taken.
NRØW	I	A	Number of rows in partition to be printed/punched, must be less than or equal to MAXRØW.

Variable	Input/ Output	Argument/ Common	Definition
NCØL	I	A	Number of columns in partition to be printed/punched.
LABEL	I	A	For PRPART and PRCØRR an NCØL-vector of Hollerith labels for printing. For PUNCØR, a one-word Hollerith label for the matrix to be punched.

Local Variables: None

Subroutines Called: None

Calling Subroutines: CØRREL, STMPR, MEASPR, GUIDE

Common Blocks: None

Logic Flow: None

3.3.34 Subroutine: PRSDEV (SDCØR, MAXRØW, NRØW, LABEL)

Entry Points: PUNSD

Purpose: To print (PRSDEV) or punch (PUNSD) a matrix of standard deviations and correlation coefficients.

Remarks: The input matrix (SDCØR) may represent a complete covariance or any diagonal sub-block thereof. It is assumed to have standard deviations on the diagonal and correlation coefficients in the upper triangle. The lower triangle is ignored. For further remarks on locating the partition to be printed/punched, see Section 3.3.33, Subroutine PRPART under Remarks.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
SDCØR	I	A	First word of partition to be printed/punched.
MAXRØW	I	A	Total number of rows in matrix from which partition is taken.
NRØW	I	A	Number of rows in partition.
LABEL	I	A	PRSDEV - an NRØW-vector of Hollerith labels corresponding to the variables in the partitions. PUNSD - a one work Hollerith label for the matrix partition.

Local Variables: None

Subroutines Called: None

Calling Subroutines: CØRREL, GUIDE, RELCØV

Common Blocks: None

Logic Flow: None

3.3.35 Subroutine: RELCØV (PIN, DXDK, EIGEN, PREL)

Purpose: To compute S/C state uncertainties relative to ephemeris body.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
PIN	I	A	Augmented covariance matrix.
DXDK	I	A	Keplerian to cartesian transformation for ephemeris body.
EIGEN	I	A	Logical flag. = T, compute eigenvectors and eigenvalues of relative covariance. = F, compute standard deviations and correlation coefficients only.
PREL	Ø	A	Relative covariance matrix.
AUGLAB	I	C	Array of augmented parameter Hollerith labels.
FØP	I	C	Final off-diagonal annihilation value for position eigenvalue computation.
FØV	I	C	Final off diagonal annihilation value for velocity eigenvalue computation.
IEPHEM	I	C	Ephemeris body coordinate system flag.
LIST	I	C	List of augmented parameter numbers.
LISTPH	I	C	List of ephemeris element parameter numbers.
NAUG	I	C	Length of augmented state vector.

Local Variables:

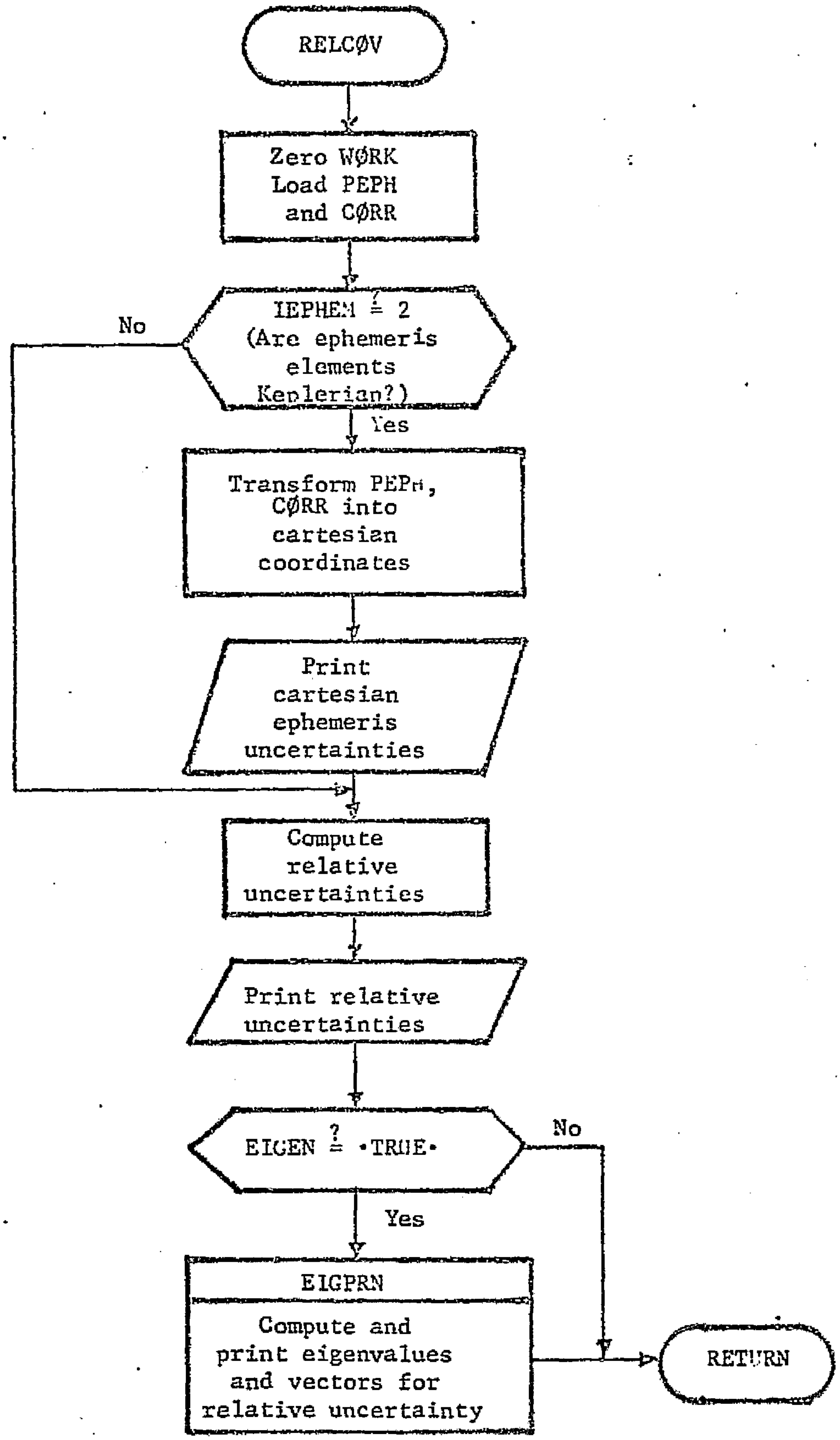
Variable	Definition
CØRR	Cross covariance of S/C state with ephemeris body cartesian state.
PEPH	Covariance of ephemeris body cartesian state.

Subroutines Called: ZERØM, LØADRC, LØDCØL, MMABT, CØPY, MMAB, SYMTRZ, VARSD, PRSDEV, MPAK, ADD, SUB, SUBT, EIGPRN

Calling Subroutines: MEASPR, GUIDE, SETEVN

Common Blocks: WØRK, (BLANK), CØNST, DIMENS, KEPCØN, LABEL, LØCATE, MEASI

Logic Flow:


RELCØV
Zero WØRK
Load PEPH
and CØRR
IEPHEM ? 2
(Are ephemeris
elements
Keplerian?)
No
Yes
Transform PEPH,
CØRR into
cartesian
coordinates
Print
cartesian
ephemeris
uncertainties
Compute
relative
uncertainties
Print relative
uncertainties
EIGEN ? .TRUE.
No
Yes
EIGPRN
Compute and
print eigenvalues
and vectors for
relative uncertainty
RETURN

3.3.36 Subroutine: SCHED (TLAST, TEVENT, DELT, JEVENT)

Purpose: To schedule for GØDSEP the next measurement or event to be processed.

Remarks: During normal operation, SCHED returns a pre-computed measurement or event and then computes and stores locally the next measurement or event to be processed. Therefore, two successive calls are required to initialize both the measurement and event scheduling sequences. The purpose in pre-computing times and event codes is to minimize search time. When a measurement is scheduled, only measurements need be scanned for the next scheduling, not events. The reverse, of course, is true when an event is scheduled.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
TLAST	I	A	Time of previous measurement/event.
TEVENT	O	A	Time of new measurement/event.
DELT	O	A	Time difference between previous and new measurement/event.
JEVENT	O	A	Integer code of new measurement/event corresponding to time TEVENT.
BIG	I	C	An awfully large number.

Variable	Input/ Output	Argument/ Common	Definition
MCØDE	I	C	Array of measurement codes to be scheduled.
MCØUNT	I/O	C	Measurement counter.
NCNTE	I/O	C	Eigenvector event counter.
NCNTG	I/O	C	Guidance event counter.
NCNTP	I/O	C	Prediction event counter.
NCNTT	I/O	C	Thrust event counter.
NEIGEN	I	C	Total number of eigenvector events.
NGUID	I	C	Total number of guidance events.
NPRED	I	C	Total number of prediction events.
NSCHED	I	C	Number of schedule times in SCHEDM to be scanned for next measurement or propagation event.
NTHRST	I	C	Total number of thrust events.
SCHEDM	I	C	Array of measurement schedule times SCHEDM(1,I) = Next time to be scheduled for measurement type MCØDE(I). SCHEDM(2,I) = Stop time for MCØDE(I). SCHEDM(3,I) = Time increment for scheduling MCØDE(I).
TEIGEN	I	C	Array of eigenvector event times.

Variable	Input/ Output	Argument/ Common	Definition
TFINAL	I	C	Final time.
TGUID	I	C	Array of guidance event times.
TPRED	I	C	Array of prediction event times.
TTHRST	I	C	Array of thrust event times.

Local Variables:

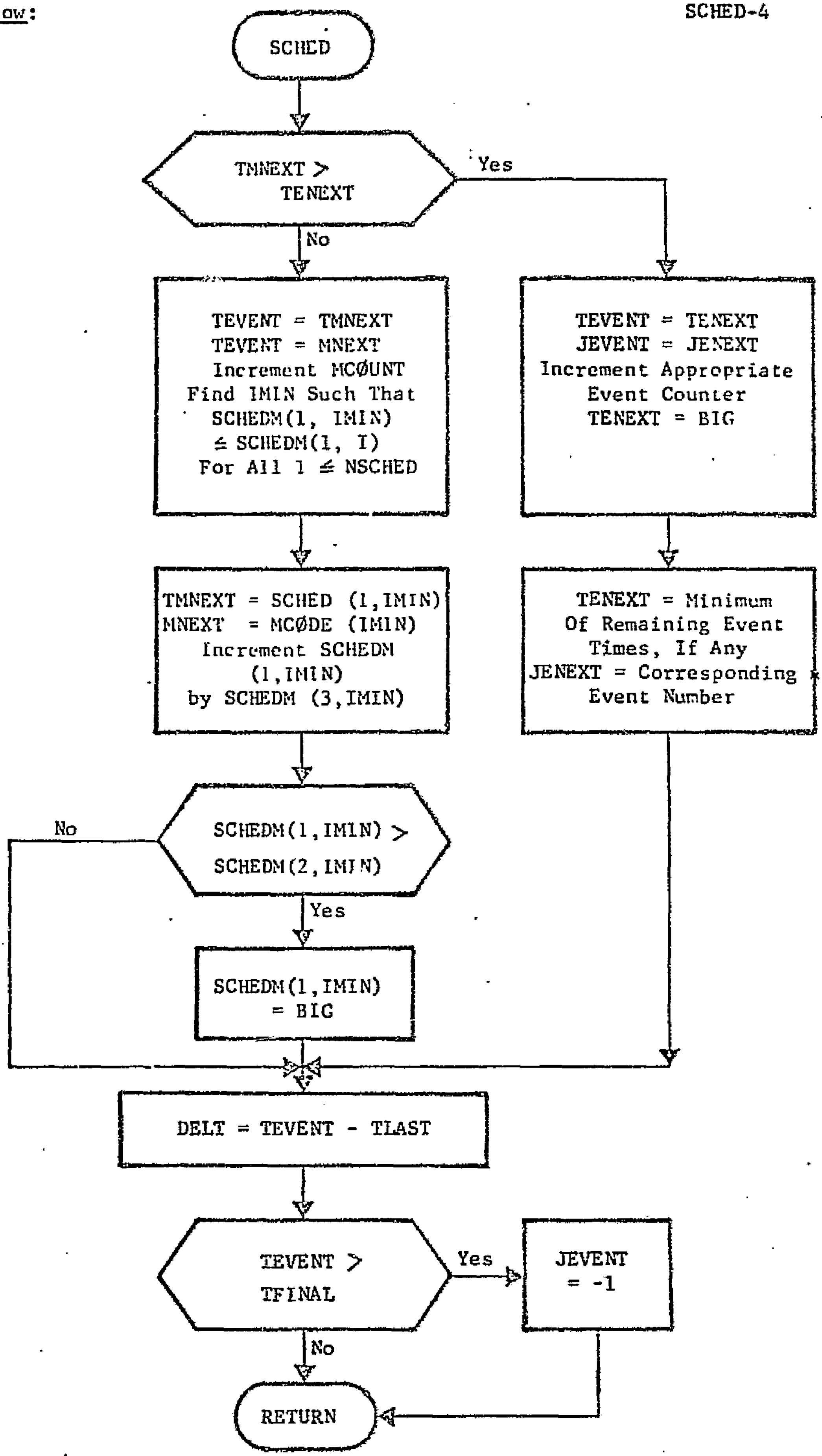
Variable	Definition
JENEXT	Integer code of next event to be scheduled.
MNEXT	Integer code of next measurement to be scheduled.
TENEXT	Time of next event to be scheduled.
TMNEXT	Time of next measurement to be scheduled.

Subroutines Called: None

Calling Subroutines: ØUTPTG, STMGEN, GØDSEP

Common Blocks: CØNST, SCHEDI, SCHEDR

Logic Flow:


SCHED
TMNEXT > TENEXT
Yes
No
TEVENT = TMNEXT
TEVENT = MNEXT
Increment MCØUNT
Find IMIN Such That
SCHEDM(1, IMIN)
≤ SCHEDM(1, I)
For All 1 ≤ NSCHED
TEVENT = TENEXT
JEVENT = JENEXT
Increment Appropriate
Event Counter
TENEXT = BIG
TMNEXT = SCHED (1,IMIN)
MNEXT = MCØDE (IMIN)
Increment SCHEDM
(1,IMIN)
by SCHEDM (3,IMIN)
TENEXT = Minimum
Of Remaining Event
Times, If Any
JENEXT = Corresponding
Event Number
SCHEDM(1,IMIN) >
SCHEDM(2,IMIN)
No
Yes
SCHEDM(1,IMIN)
= BIG
DELT = TEVENT - TLAST
TEVENT >
TFINAL
Yes
No
JEVENT
= -1
RETURN

3.3.37 Subroutine: SETEVN

Purpose: Event print control and propagation control for prediction events.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
MESEVN	I	C	Event code. = 1, propagation. = 2, eigenvector. = 3, thrust switching. = 4, guidance. = 5, prediction.
AUGLAB	I	C	Array of augmented parameter Hollerith labels.
DXDKST	I	C	Keplerian to cartesian transformation for ephemeris body.
EVLAB	I	C	Hollerith event label array.
FØP	I	C	Final off-diagonal annihilation value for position eigenvalue computation.
IPRØP	I	C	Print control flag for propagation events. = 0, no print = 1, print standard deviations and correlation coefficients for S/C state only = 2, full eigenvector event print.
NAUG	I	C	Length of augmented state vector.
NCNTP	I	C	Number of current prediction event.
NEPHEL	I	C	Number of ephemeris elements in augmented state vector.

Variable	Input/ Output	Argument/ Common	Definition
P	I	C	Location in blank common of current knowledge covariance.
PLAB	I	C	Array of Hollerith labels for knowledge covariance sub-blocks.
PLØCAL	I	C	Location in blank common of working storage provided to subroutine RELCØV.
PTEMP	I	C	Location in blank common of predicted knowledge covariance.
SCMASS	I	C	Current S/C mass.
TCURR	I	C	Current trajectory time.
TDUR	Ø	C	Maximum integration time (seconds) for TRAJ.
TFINAL	I	C	Error analysis final time.
TGSTØP	I	C	Maximum integration time (days) if prediction event requires integration past TFINAL.
TM	I	C	Conversion constant, seconds/day.

Local Variables:

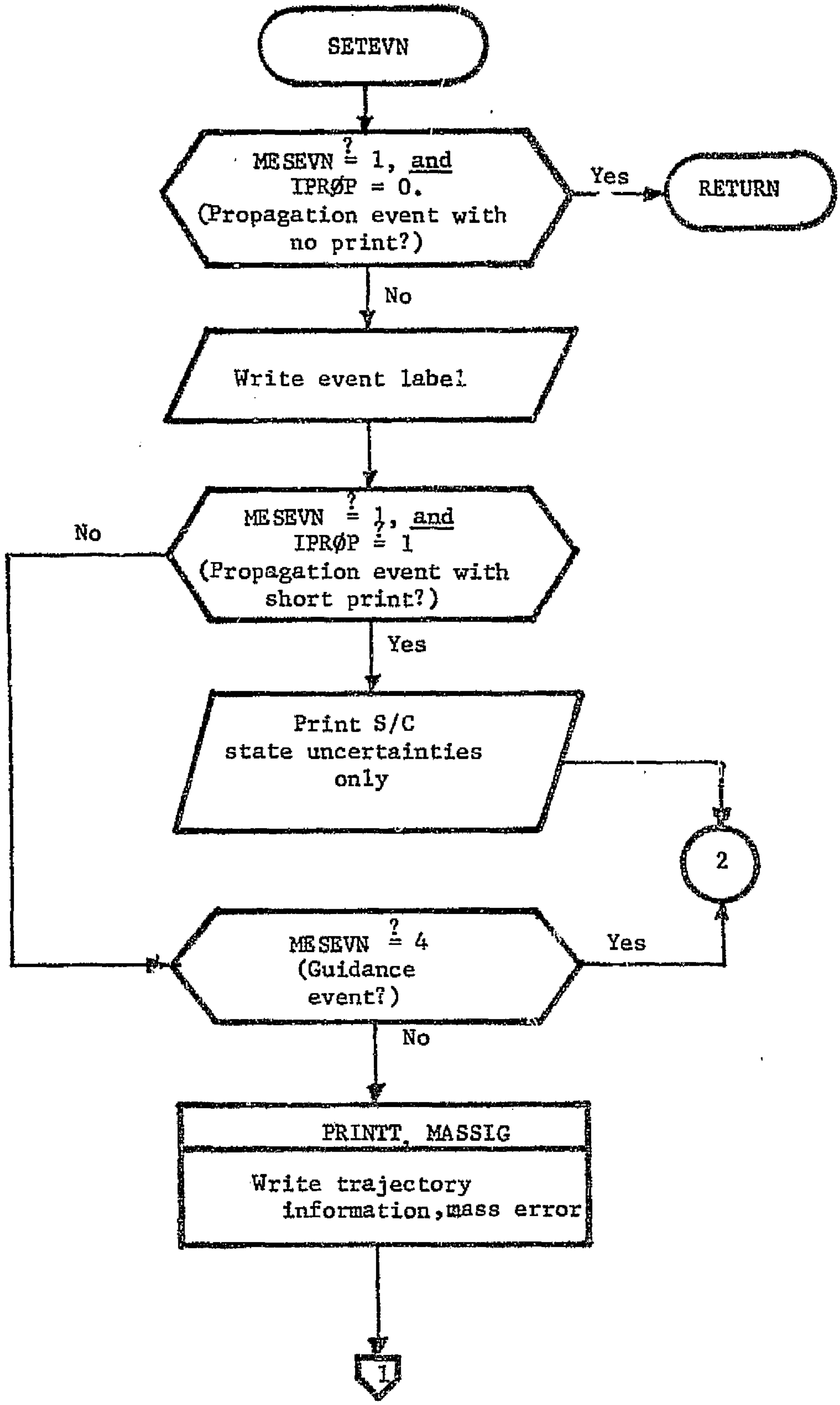
Variable	Definition
LP	Location in blank common of covariance to be operated on by RELCØV and CØRREL.

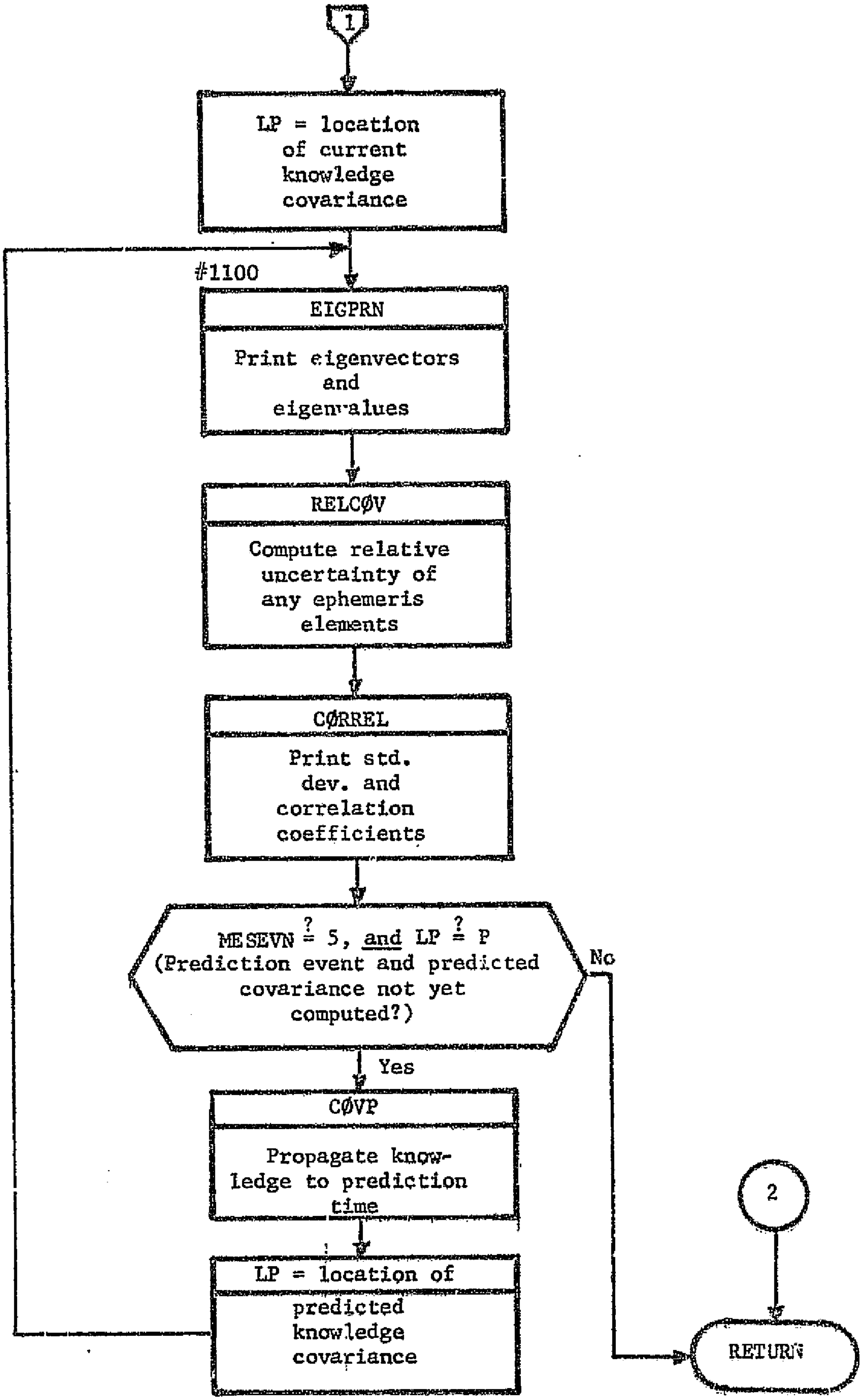
Subroutines Called: JØBTLE, MPAS, VARSD, PRSDEV, PRINTT, EIGPRN, RELCØV, CØRREL, CØVP, MASSIG, DYNØ

Calling Subroutine: GØDSEP

<u>Common Blocks</u>: WØRK, (BLANK), CØNST, DIMENS, GUIDE, KEPCØN, LABEL, LØCATE, LØGIC, MEASI, PRØPI, SCHEDI, SCHEDR, TIME, TRAJ1

Logic Flow:


SETEVN
MESEVN ≟ 1, and
IPRØP = 0.
(Propagation event with
no print?)
Yes
RETURN
No
Write event label
MESEVN ≟ 1, and
IPRØP ≟ 1
(Propagation event with
short print?)
No
Yes
Print S/C
state uncertainties
only
2
MESEVN ≟ 4
(Guidance
event?)
Yes
No
PRINTT, MASSIG
Write trajectory
information, mass error
1


1
LP = location of current knowledge covariance
#1100
EIGPRN
Print eigenvectors and eigenvalues
RELCØV
Compute relative uncertainty of any ephemeris elements
CØRREL
Print std. dev. and correlation coefficients
MESEVN ≟ 5, and LP ≟ P
(Prediction event and predicted covariance not yet computed?)
No
Yes
CØVP
Propagate knowledge to prediction time
LP = location of predicted knowledge covariance
2
RETURN

3.3.38 Subroutine: SETGUI

Purpose: Set up control for guidance event. Performs all computations which must be done in primary overlay which consists primarily of interfacing with TRAJ.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
BIG	I	C	Enormous constant, 1.E20
BURNP	O	C	Mass and thrust at guidance start and stop
CHEKPR(8)	I	C	Logical flag. = T, generate transition matrices for guidance by reading STM file. = F, integrate transition matrices for guidance in TRAJ.
DELAY	Ø	C	Guidance delay time for current event.
DXDKAF	Ø	C	DXDKST evaluated at end of burn interval.
DXDKBR	Ø	C	DXDKST evaluated at beginning of burn interval.
DXDKST	I	C	Keplerian to cartesian ephemeris transformation from STMRDR, corresponds to beginning of guidance delay interval.
GT	I/Ø	C	Transformation matrix for subroutine DYNØ evaluated at end of propagation interval.
GTBURN	Ø	C	GT matrix evaluated at beginning of burn interval.

Variable	Input/ Output	Argument/ Common	Definition
GTDLAY	Ø	C	GT matrix evaluated at beginning of delay interval.
GTØFF	Ø	C	GT matrix evaluated at end of burn interval.
GTSAVE	Ø	C	GT matrix evaluated at beginning of current propagation interval for subroutine DYNØ.
IAUGDC	I/O	C	Dynamic parameter augmentation flags.
ICALL	Ø	C	Setup parameter for TRAJ (Section 3.5)
IEP	I	C	Set UP, VP below.
IEPHEM	I	C	Ephemeris element coordinate system flag.
IGPØL	I	C	Array of guidance policy flags.
IGREAD	I	C	Array of namelist $GEVENT read control flags.
INTEG	Ø	C	Setup parameter for TRAJ (Section 3.5)
IPØL	Ø	C	Guidance policy flag for current event.
IPRINT	Ø	C	Setup parameter for TRAJ (Section 3.5)
IREAD	Ø	C	$GEVENT read policy for current event.
ISTØP	Ø	C	Stopping condition parameter for TRAJ (Section 3.5)
KUTØFF	Ø	C	Flag indicating actual integrator stopping conditions.
LISTDY	I	C	List of dynamic parameters contained in transition matrix generated either from STM file or TRAJ.

Variable	Input/ Output	Argument/ Common	Definition
LØCTC	I	C	Location in blank common of transition matrix returned by TRAJ.
MEVENT	Ø	C	Setup flag for TRAJ (Section 3.5)
NAUG	I	C	Length of augmented state vector.
NCNTG	I	C	Number of current guidance event.
NPHSTM	I	C	Dimension of transition matrix returned by subroutine STMRDR or by TRAJ.
NPRI	I	C	Body number of primary integration body.
NTPHAS	I	C	Number of current thrust phase.
PG1	I	C	Locations in blank common of working storage for guidance related covariance computations.
PG2	I	C	
PHI	I	C	Location in blank common of transition matrix.
PLØCAL	I	C	Location in blank common of covariance working storage.
PTEMP	I	C	Location in blank common of covariance working storage.
S	Ø	C	Guidance sensitivity matrix, cutoff state wrt controls.
SCMASS	I	C	S/C mass.
SMASS	I	C	Mass of sun.
STATEO	Ø	C	Initial integration state for TRAJ.
TBURN	Ø	C	Length of burn interval for current event.

Variable	Input/ Output	Argument/ Common	Definition
TCUTØF	I	C	Array of guidance event cutoff times.
TDELAY	Ø	C	Guidance delay time for current event.
TDUR	Ø	C	Maximum integration time (seconds) for TRAJ.
TEVNT	Ø	C	Event time for TRAJ.
TFINAL	I	C	Error analysis final time.
TGSTØP	I	C	Maximum integration time if guidance event needs transition matrices evaluated past final time.
TGUID	I	C	Array of guidance event scheduled times.
THRACC	I	C	Thrust acceleration vector.
TIMFTA	I	C	Target condition evaluation time for fixed time of arrival guidance.
TM	I	C	Conversion constant, seconds/ day.
TØFF	Ø	C	Cutoff time for current event.
TØN	Ø	C	Maneuver execution time for current event.
TREF	Ø	C	TRAJ reference time for integration initialization.
TSTM	I	C	STM file time.
UP (1,IEP)	I	C	Position of ephemeris body.
VP (1,IEP)	I	C	Velocity of ephemeris body.
UTRUE	I	C	S/C heliocentric ecliptic position vector used to define STATEO for TRAJ initialization.

Variable	Input/ Output	Argument/ Common	Definition
VTRUE	I	C	S/C heliocentric ecliptic velocity vector used to define STATEO for TRAJ initialization.
VRNIER	Ø	C	Logical flag. = T, current maneuver is vernier = F, current maneuver is primary.

Local Variables:

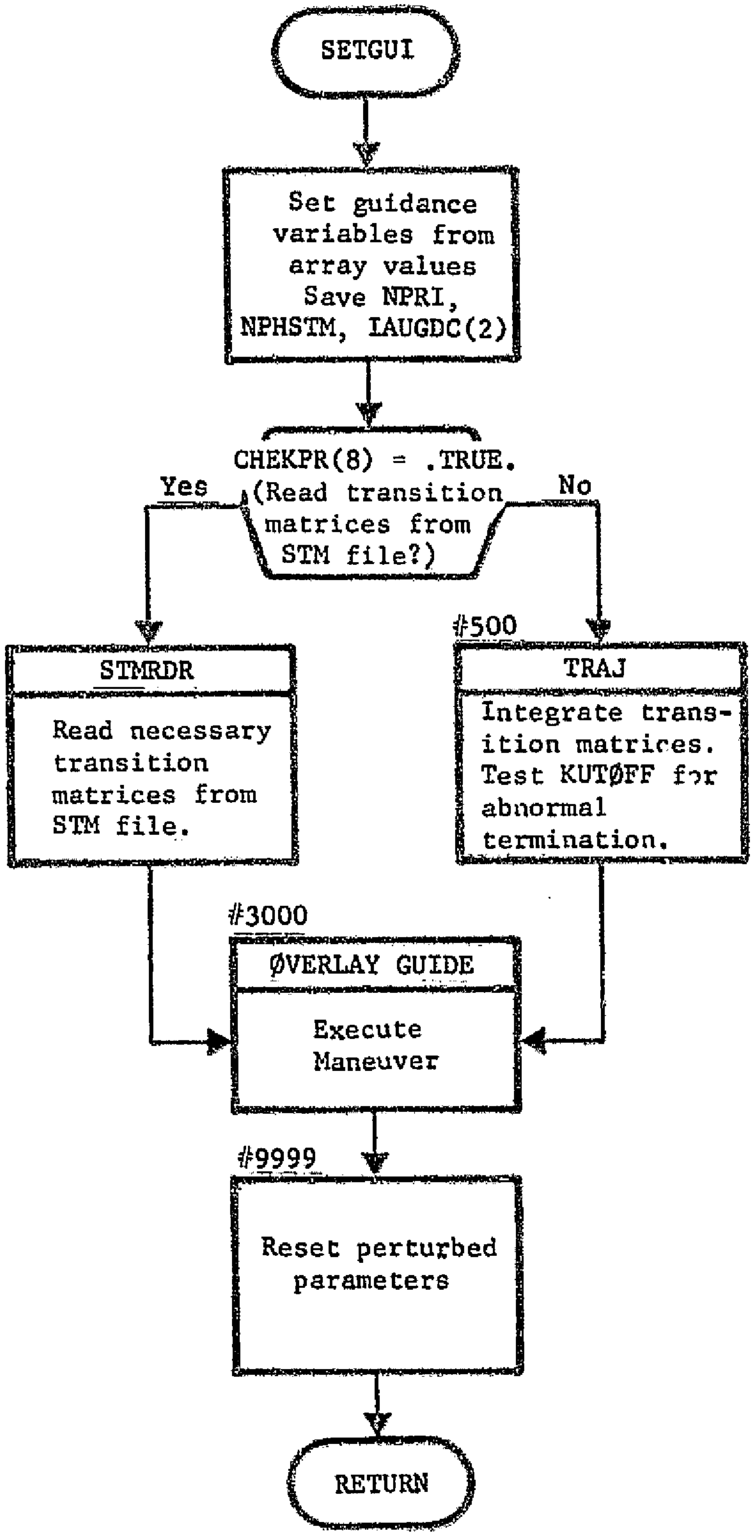
Variable	Definition
IHOLD1, IHOLD2, IHOLD3 IHOLD4, IHOLD5, IHOLD6	Locations for saving parameter values which will be changed by calls to either STMRDR or TRAJ.
TSTMSV	Saves STM file time (TSTM) when generating state transition matrices by calling STMRDR.

Subroutines Called: CØPY, ZERØM, STMRDR, MPAK, STMUSE, STMPR, PARKEP, BØMB, JØBTLE

Calling Subroutine: GØDSEP

Common Blocks: WØRK, (BLANK), CØNST, DIMENS, EPHEM, GUIDE, KEPCØN, LØCATE, LØGIC, MEASI, PRØPI, PRØPR, SCHEDI, SCHEDR, TIME, TRAJ1, TRAJ2

Logic Flow:


SETGUI
Set guidance variables from array values Save NPRI, NPHSTM, IAUGDC(2)
CHEKPR(8) = .TRUE. (Read transition matrices from STM file?)
Yes
No
STMRDR
Read necessary transition matrices from STM file.
#500
TRAJ
Integrate transition matrices. Test KUTØFF for abnormal termination.
#3000
ØVERLAY GUIDE
Execute Maneuver
#9999
Reset perturbed parameters
RETURN

3.3.39 Subroutine: STMGEN

Purpose: Generate STM file.

Remarks: For effective process noise computation subroutine DYNØ requires the evaluation at beginning and end of a propagation interval of the rotation matrix from body-centered magnitude, cone, clock system to heliocentric ecliptic cartesian coordinates. This transformation must be saved on the STM file. At thrust phase change two such transformations are required, one for each phase evaluated at the same time point. Calls to the trajectory overlay are generated to guarantee that this transformation is always evaluated for the interval just ending, and an extra call to subroutine EP is required to evaluate the transformation at the beginning of the new thrust phase. This pertains to statements between statement numbers 300 and 400.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
CHEKPR(1)	I	C	Check print flag. = T, write to output all trajectory information written on STM file. = F, no write to output.
DELTIM	I	C	Time difference between previously and currently scheduled events.

Variable	Input/ Output	Argument/ Common	Definition
LØCM	I	C	Location as blank common of current S/C mass.
LØCTC	I	C	Location as blank common of current transition matrix.
MESEVN	I	C	Current event code.
NCNTT	I	C	Number of current thrust event.
NPHSTM	I	C	Dimension of transition matrix.
TCURR	I	C	Currently scheduled trajectory time.
TFINAL	I	C	Stop time for STM file generation.
TM	I	C	Conversion constant, seconds/ day.
TPAST	I	C	Previously scheduled trajectory time.
INTEG ISTØP ICALL MEVENT TREF, TEVNT	Ø	C	Initialization parameters for TRAJ.
TCURR, TPAST NPRI, NTPHAS APERT, APRIM CØMMØN (LØCM) RPACC, THRACC UP, VP, UREL URELM, VREL VRELM, UTRUE	Ø	C	Trajectory information written to STM file. See common block descriptions for individual variable descriptions.

Variable	Input/ Output	Argument/ Common	Definition
VTRUE, UTRUEM, VTRUEM, WPØWER, GT, GTSAVE, CØMMØN (LØCTC)	Ø	C	Trajectory information written to STM file. See common block descriptions for individual variable descriptions.

Local Variables: None

Subroutines Called: CØPY, SCHED, EP

Calling Subroutine: GØDSEP

Common Blocks: WØRK, (BLANK), CØNST, DIMENS, LØGIC, PRØPR, SCHEDI, SCHEDR, TIME, TRAJ1, TRAJ2

Logic Flow: None

3.3.40 Subroutine: STMPR (T, TF, PHIMAT)

Purpose: To print state transition matrix partitions and effective process noise covariance if computed.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
T	I	A	Trajectory time at beginning of propagation interval.
TF	I	A	Trajectory time at end of propagation interval.
PHIMAT	I	A	Augmented transition matrix over propagation interval.
AUGLAB	I	C	Array of augmented parameter Hollerith labels.
DYNOIS	I	C	Dynamic noise flag.
LØCAUG	I	C	Array locating sub-blocks within augmented transition matrix.
LØCLAB	I	C	Array locating state vector partions within AUGLAB array.
NAUG	I	C	Length of augmented state vector.
NDIM	I	C	Array of lengths of individual state vector partitions.
PRNSTM	I	C	Output control flag determining sets of transition matrix sub-blocks to be printed. = T, print sensitivities of relevant state vector partition to entire augmented state. = F, no sensitivities printed for relevant state vector partition.

Variable	Input/ Output	Argument/ Common	Definition
			(1) - S/C state (2) - Solve-for parameters (3) - Dynamic consider parameters (4) - Measurement consider parameters (5) - Ignore parameters.
Q	I	C	Effective process noise covariance.
VECLAB	I	C	Array of state vector partition Hollerith labels.

Local Variables: None

Subroutines Called: PRPART, MATØUT

Calling Subroutines: MEASPR, STMRDR, GUIDE, SETGUI

Common Blocks: WØRK, DIMENS, LABEL, LØGIC, PRØPR

Logic Flow: None

3.3.41 Subroutine: STMRDR (T, TF, IØPT)

Purpose: To read transition matrices and trajectory information from STM file (TAPE 3).

Remarks: During STM file creation the user should have scheduled as fine a time grid of trajectory points as will ever be necessary for the particular mission. Therefore, situations will occur during STM file reading where many time points are encountered on the file between time points requested by the scheduler for the current error analysis. In this situation transition matrices over the short time intervals are chained to produce the required transition matrix over the complete time interval.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
T	I	A	Trajectory time at beginning of propagation interval.
TF	I	A	Scheduled trajectory time at end of propagation interval.
IØPT	I	A	Option flag. = 0, normal read. = +1, count number of records read for future backspace capability. = -1, same as +1 but compute guidance sensitivity matrix in addition.
CHEKPR(1)	I	C	Check print flag.

Variable	Input/ Output	Argument/ Common	Definition
			= T, print all trajectory information read from STM file and all intermediate products in transition matrix chaining. = F, no print.
DELTIM	I/O	C	Input as scheduled interval length. If STM file is already positioned within forward tolerance DELTIM is set to 0.
DXDKST	Ø	C	Keplerian to cartesian transformation for ephemeris body evaluated at time TSTM.
IEP	I	C	See UP, VP below.
IEPHEM	I	C	Ephemeris coordinate system flag.
LISTDY	I	C	List of dynamic parameters included in transition matrix read from STM file.
MESH	Ø	C	Logical flag. = T, successful mesh of scheduled trajectory times with STM file times. = F, unsuccessful mesh.
NAUG	I	C	Length of augmented state vector.
NPHSTM	I	C	Dimension of transition matrix read from STM file.
PHI	I	C	Location in blank common of output transition matrix.
PLOCAL	I	C	Location in blank common of transition matrix working storage for chaining.

Variable	Input/ Output	Argument/ Common	Definition
PTEMP	I	C	Location in blank common of transition matrix working storage for chaining.
S	Ø	C	Guidance sensitivity matrix computed if IØPT = 1.
SCHFTL	I	C	Logical flag. = T, failure to mesh is fatal. = F, failure to mesh is not fatal.
SMASS	I	C	Mass of sun.
TØLBAK	I	C	Backward tolerance on file time meshing.
TØLFØR	I	C	Forward tolerance on file time meshing.
TSTM	Ø	C	Current STM file time.
UP(1,IEP)	Ø	C	Heliocentric position of ephemeris body.
VP(1,IEP)	Ø	C	Heliocentric velocity of ephemeris body.
NPRI, NTPHAS APERT, APRIM SCMASS, RPACC THRACC,UP,VP UREL, URELM VREL, VRELM UTRUE, VTRUE UTRUEM, VTRUEM WPØWER, GT GTSAVE	Ø	C	Trajectory related information read from STM file. See individual parameter definitions in common block descriptions.

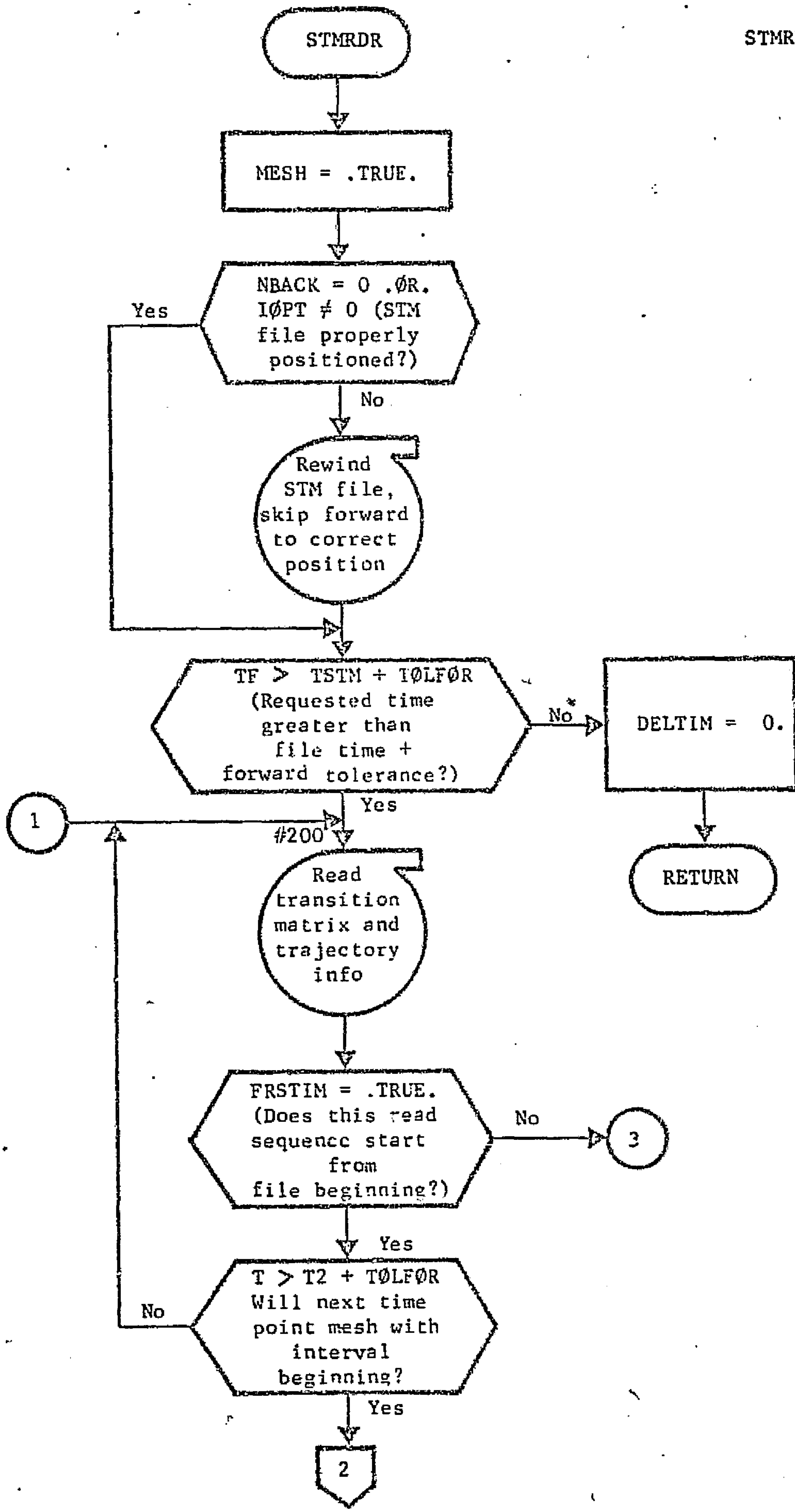
Local Variables:

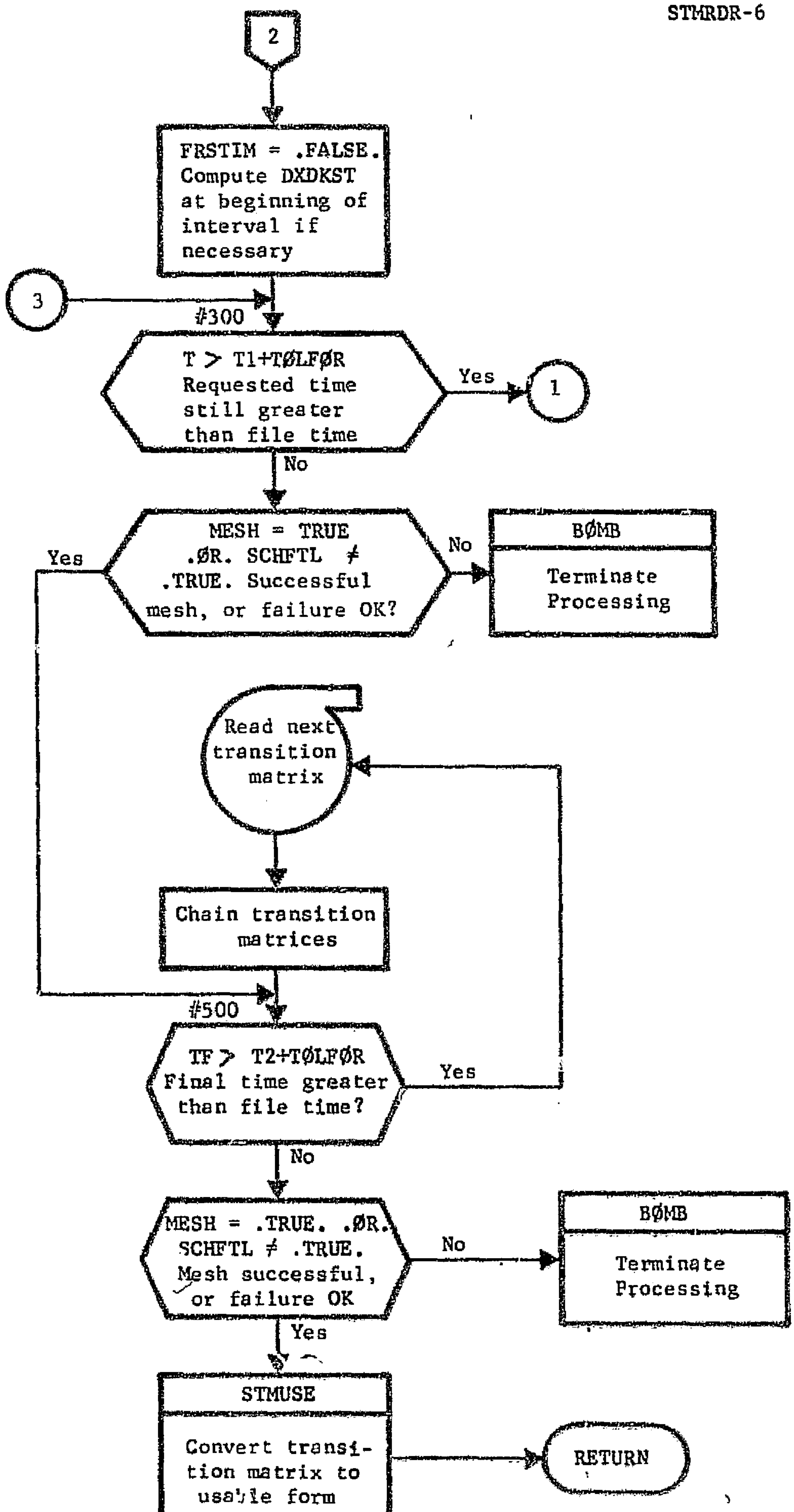
Variable	Definition
IHOLD	Intermediate holding variable used when exchanging values of IPHI2 and IPHI3.
IPHI2, IPHI3	Initially set to PLØCAL and PTEMP respectively. Values are switched to avoid copying of intermediate transition matrices used in chaining.
NBACK	Number of records read when IØPT = 0 to be used for backspacing.
NUPPER	Upper word limit for reading STM record.
TSTMO	Last Value of TSTM when IØPT = 0.

Subroutines Called: VECMAG, PARKEP, BØMB, MMAB, MATØUT, MPAK, STMUSE, STMPR

Calling Subroutines: CØVP, SETGUI

Common Blocks: WØRK, (BLANK), CØNST, DIMENS, EPHEM, GUIDE, KEPCØN, LØCATE, LØGIC, MEASI, PRØPR, SCHEDI, SCHEDP, TIME, TRAJ1, TRAJ2


STMRDR
MESH = .TRUE.
NBACK = 0 .ØR. IØPT ≠ 0 (STM file properly positioned?)
Yes
No
Rewind STM file, skip forward to correct position
TF > TSTM + TØLFØR (Requested time greater than file time + forward tolerance?)
No*
DELTIM = 0.
RETURN
Yes
1
#200
Read transition matrix and trajectory info
FRSTIM = .TRUE. (Does this read sequence start from file beginning?)
No
3
Yes
T > T2 + TØLFØR Will next time point mesh with interval beginning?
No
Yes
2


2
FRSTIM = .FALSE. Compute DXDKST at beginning of interval if necessary
3
#300
T > T1+TØLFØR Requested time still greater than file time
Yes
1
No
MESH = TRUE .ØR. SCHFTL ≠ .TRUE. Successful mesh, or failure OK?
Yes
No
BØMB
Terminate Processing
Read next transition matrix
Chain transition matrices
#500
TF > T2+TØLFØR Final time greater than file time?
Yes
No
MESH = .TRUE. .ØR. SCHFTL ≠ .TRUE. Mesh successful, or failure OK
No
BØMB
Terminate Processing
Yes
STMUSE
Convert transition matrix to usable form
RETURN

3.3.42 <u>Subroutine</u>: STMUSE (THRNUM, DXDK, STMIN, NIN, LISTIN, STMØUT, NØUT)

<u>Purpose</u>: To convert state transition matrix as read from STM file to state transition matrix as needed by augmented covariance matrix.

<u>Remarks</u>: There are four possible operations required to convert STM file transition matrices to the augmented transition matrix required for covariance propagation:

(1) ordering of rows and columns with insertions for measurement parameters and deletions for unused dynamic parameters as necessary

(2) scaling of thrust parameter sensitivities to account for number of thruster operating over current phase;

(3) the setting to identity of ephemeris element to ephemeris element sensitivity sub-block if stationary cartesian or Keplerian elements are augmented; and

(4) the coordinate transformation of S/C state sensitivities to ephemeris elements if those elements are Keplerian.

<u>Input/Output</u>:

Variable	Input/ Output	Argument/ Common	Definition
THRNUM	I	A	Number of thrusters operating over transition matrix interval.

Variable	Input/ Output	Argument/ Common	Definition
DXDK	I	A	Keplerian to cartesian transformation for ephemeris elements evaluated at beginning of interval.
STMIN	I	A	Input transition matrix.
NIN	I	A	Dimension of input transition matrix.
LISTIN	I	A	List of parameters included in input transition matrix.
STMØUT	Ø	A	Output transition matrix.
NØUT	Ø	A	Dimension of output transition matrix (required only variably dimensioning STMØUT).
IEPHEM	I	C	Coordinate system flag for ephemeris elements.
LIST	I	C	Parameter list for output transition matrix.
LISTPH	I	C	Parameter list of possible ephemeris elements.
NEPHEL	I	C	Number of ephemeris elements augmented to state.

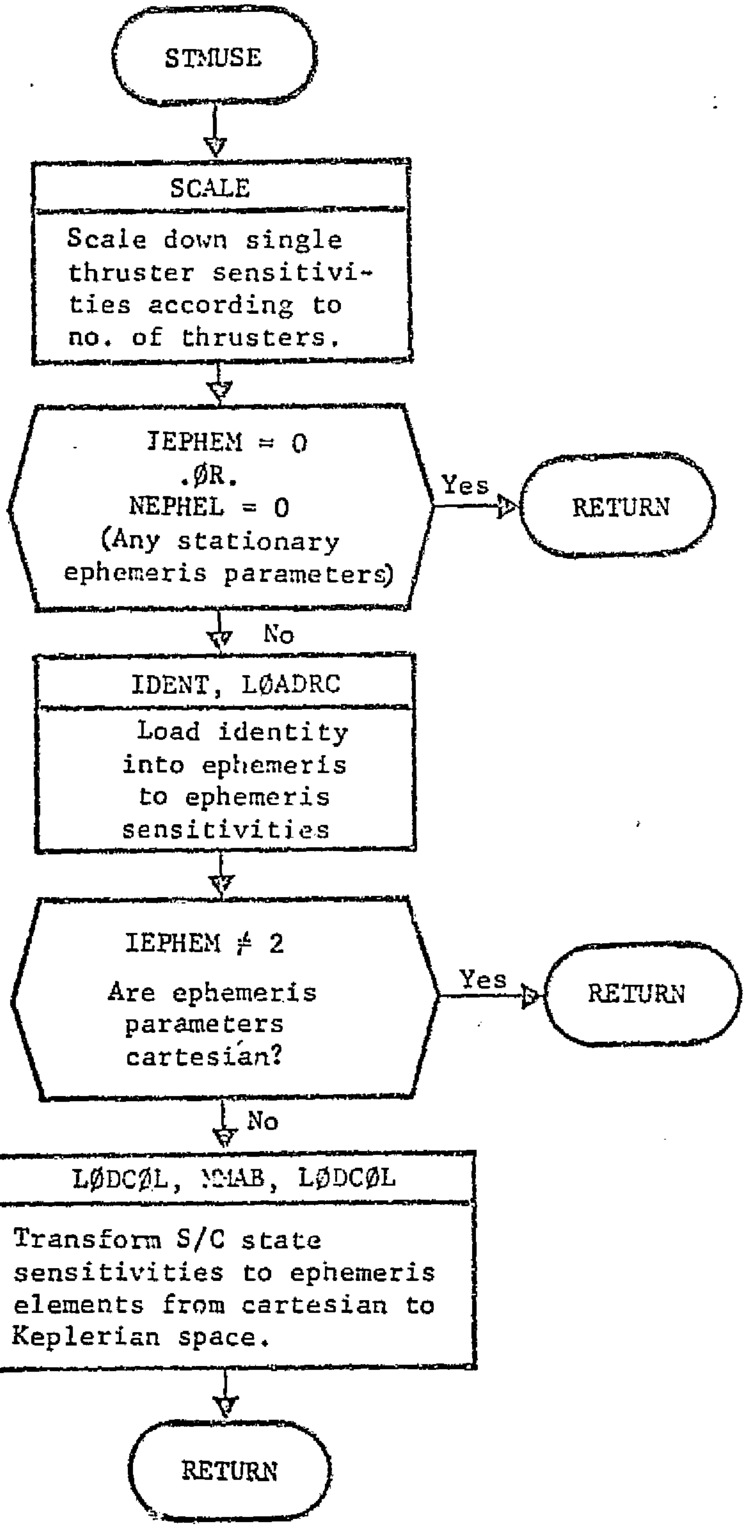
Local Variables: None

Subroutines Called: IDENT, LØADRC, SQRT, LØCLST, SCALE, LØDCØL, MMAB

Calling Subroutines: STMRDR, SETGUI

Common Blocks: WØRK, DIMENS, KEPCØN, MEASI

Logic Flow:


STMUSE
SCALE
Scale down single thruster sensitivities according to no. of thrusters.
IEPHEM = 0 .ØR. NEPHEL = 0 (Any stationary ephemeris parameters)
Yes
RETURN
No
IDENT, LØADRC
Load identity into ephemeris to ephemeris sensitivities
IEPHEM ≠ 2
Are ephemeris parameters cartesian?
Yes
RETURN
No
LØDCØL, MMAB, LØDCØL
Transform S/C state sensitivities to ephemeris elements from cartesian to Keplerian space.
RETURN

3.3.43 Subroutine: VERR (VARDV, DV, CØVERR)

Purpose: To compute the ΔV execution error covariance.

Method: Variances in ΔV proportionality, resolution and two pointing angles are applied to the input ΔV to form the execution error covariance (See Section 6.3 of the Analytic Manual).

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
VARDV	I	A	ΔV execution error variances: σ^2_{PRO}, σ^2_{RES}, σ^2_{α}, σ^2_{ϵ}
DV	I	A	$\Delta \underset{\sim}{V} = (\Delta V_x, \Delta V_y, \Delta V_z)$
CØVERR	O	A	Execution error covariance

Subroutines Called: None

Calling Subroutines: GUIDE

Common Blocks: None

Logic Flow: None

3.4 Subroutine: SIMSEP

Purpose: To control the overall logic flow of the trajectory simulation mode.

Method: SIMSEP is the main subroutine in the trajectory simulation mode. Its primary function is to control the execution of algorithms and logic according to the operation and option flags specified during input. This is done in two basic cycles within the program. The first, or outer cycle, is the so-called Monte Carlo mission cycle where a complete actual trajectory is propagated from beginning to end. Included within the mission cycle is the guidance event loop where trajectory estimation and guidance are performed to keep the "actual" trajectory on course. After many sample missions have been flown, certain statistical parameters are computed to aid in the deduction of expected trajectory characteristics and system performance.

Remarks: One of the key operations performed in SIMSEP and its subordinate routines is the propagation of trajectories from one time point to another. This operation may simultaneously include the generation of state transition matrices. Since all communications with the integrator are by common block variables, the explicit in line initialization of integrator control variables prior to calling the trajectory routine is evident throughout SIMSEP. A list of variables which must be defined to properly initialize the trajectory is given below. This list should clarify how SIMSEP's interface with TRAJ is performed.

Variable	Definition
EPØCH	Initial trajectory epoch, a Julian date.
TREF	Trajectory starting time (in seconds) measured from EPØCH.
TDUR	Trajectory termination time (in seconds) measured from EPØCH.
STATEO	State vector specified at TREF.
SCMASS	S/C mass specified at TREF.
NTPHAS	Thrust phase number of TREF.
NPRI	Primary body number at TREF.
ICALL	Trajectory initialization flag. ICALL = 1, the trajectory is initialized and propagated. ICALL = 2, the trajectory is initialized only. ICALL = 3, the trajectory is propagated from a previous integration step.
INTEG	Flag indicating which equations are to be integrated in TRAJ. INTEG = 1, equations of motion and variational equations are to be integrated.

Variable	Definition
	INTEG = 2, only the equations of motion are integrated.
ISTØP	Trajectory stopping condition flag. ISTØP = 1, the trajectory integration is ended at TDVR. ISTØP = 2, the trajectory integration is ended when closest approach is detected at the target body. ISTØP = 3, the trajectory is stopped when the sphere of influence is encountered.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
NREF	I	C	State vector read-in flag.
UREL	I	C	Relative s/c position vectors. UREL (i,1) for i = 1, 2, 3 is the heliocentric position vector of the s/c. UREL (i, ITP) for i = 1, 2, 3 is the position vector relative to the target planet.
VREL	I	C	VREL (i,1) for i = 1, 2, 3 is the heliocentric velocity vector of the s/c. VREL (i, ITP) for i = 1, 2, 3 is the velocity vector relative to the target planet.
BLANK (LØCM)	I	C	Current s/c mass at any given instant along the trajectory integration.
TSTØP	I	C	Trajectory stop time relative to EPØCH.

Variable	Input/ Output	Argument/ Common	Definition
EPØCH	I	C	Initial epoch of the mission. A Julian data corresponding to the launch of the mission.
TGE	I	C	Epoch of a guidance event.
IRAN	I	C	Random number seed.
NØISED	I	C	Thrust process noise flag. If NØISED = 1, time-varying dynamic noise is activated in the trajectory integrator. If NØISED = 0, there is no dynamic noise.
PG	I	C	Initial s/c control covariance in eigenvector/ eigenvalue form.
KTERR	I	C	Flag to indicate whether or not a trajectory is to be propagated after a given guidance correction to the designated target to evaluate target errors. If KTERR = 1, target errors are computed. If KTERR = 0, no target errors.
NSAMP	I	C	Previous number of Monte Carlo cycles that have been processed for a given guidance event.
MC	I	C	Previous number of Monte Carlo cycles that have been processed for the total mission.
RXGE	I	C	Reference trajectory state vectors at guidance events.
RMGE	I	C	Reference s/c mass at guidance events.
RXTAR	I	C	Reference trajectory state at the target time.
RMTAR	I	C	Reference s/c mass at the target time.

Variable	Input/ Output	Argument/ Common	Definition
THRUST	I	C	Thrust control array.
MTPH	I	C	Thrust control phase number at guidance events.
STHRT3	I	C	Stored thrust control array for the reference trajectory thrust profile.
NGUID	I	C	Number of guidance events for this mission.
NCYCLE	I	C	Number of Monte Carlo cycles for this SIMSEP run.

Local Variables:

Variable	Definition
IC	Monte Carlo cycle counter for complete missions.
IMAN	Guidance event counter for completed guidance events within a mission.
XREFO	Initial reference trajectory state vector.
XA	Actual trajectory state vector.
XE	Estimated trajectory state vector.
XT	Actual trajectory final target variables.
IPRNT	Print output flag.
ICNVEG	Guidance convergence flag.

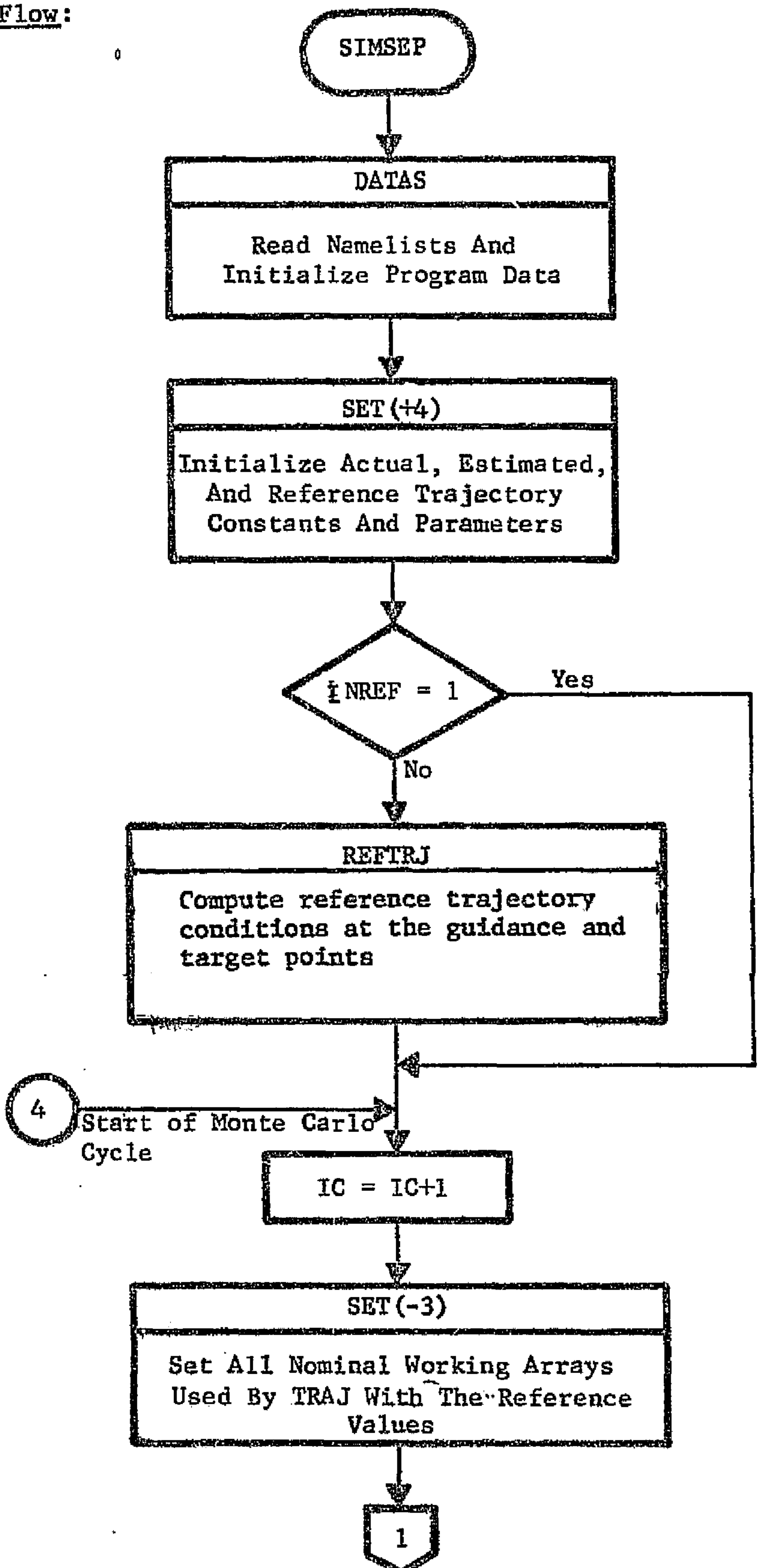
Variable	Definition
DELTAU	Guidance control corrections computed at a guidance event.
IGUID	Guidance law flag.

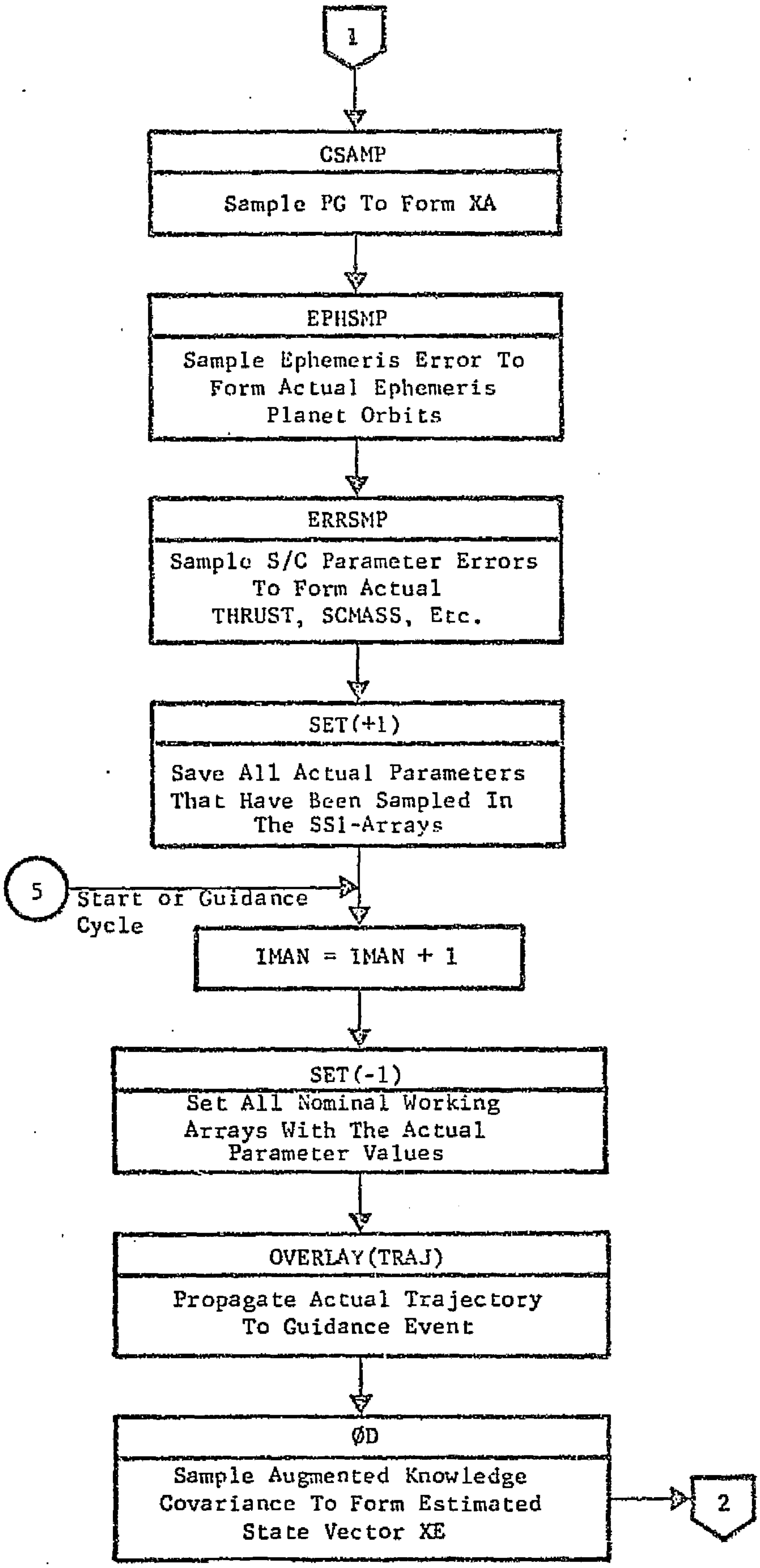
Subroutines Called: CØPY, CSAMP, DATAS, EPHSMP, ERRSMP, EXGUID, LGUID, NLGUID, NØISE, ØD, ØPSTAT, TRAJ, REFRTJ, SET, SPRNT1, STAT, TCØMP, VECMAG, ZERØM

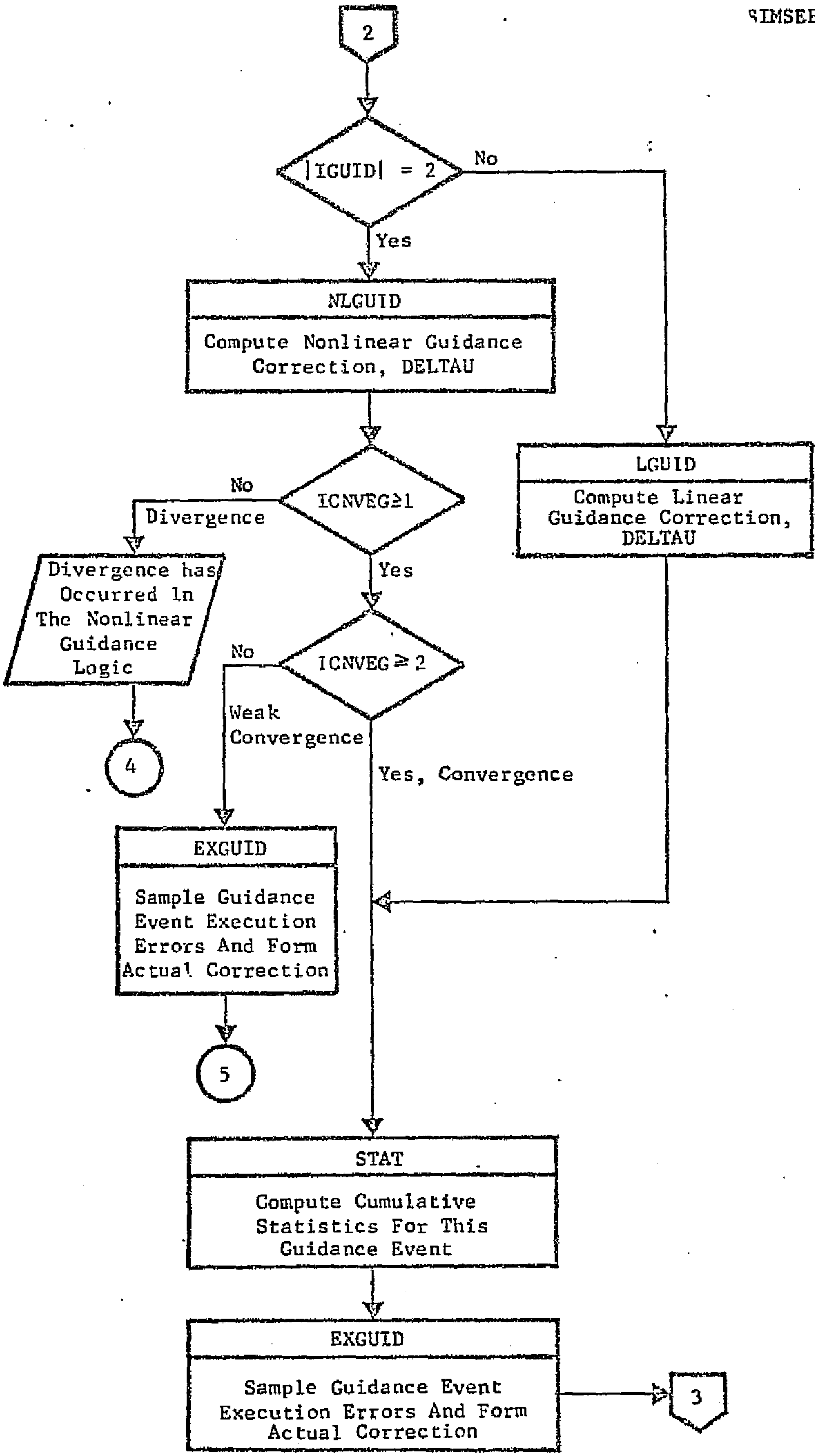
Calling Subroutines: MAPSEP

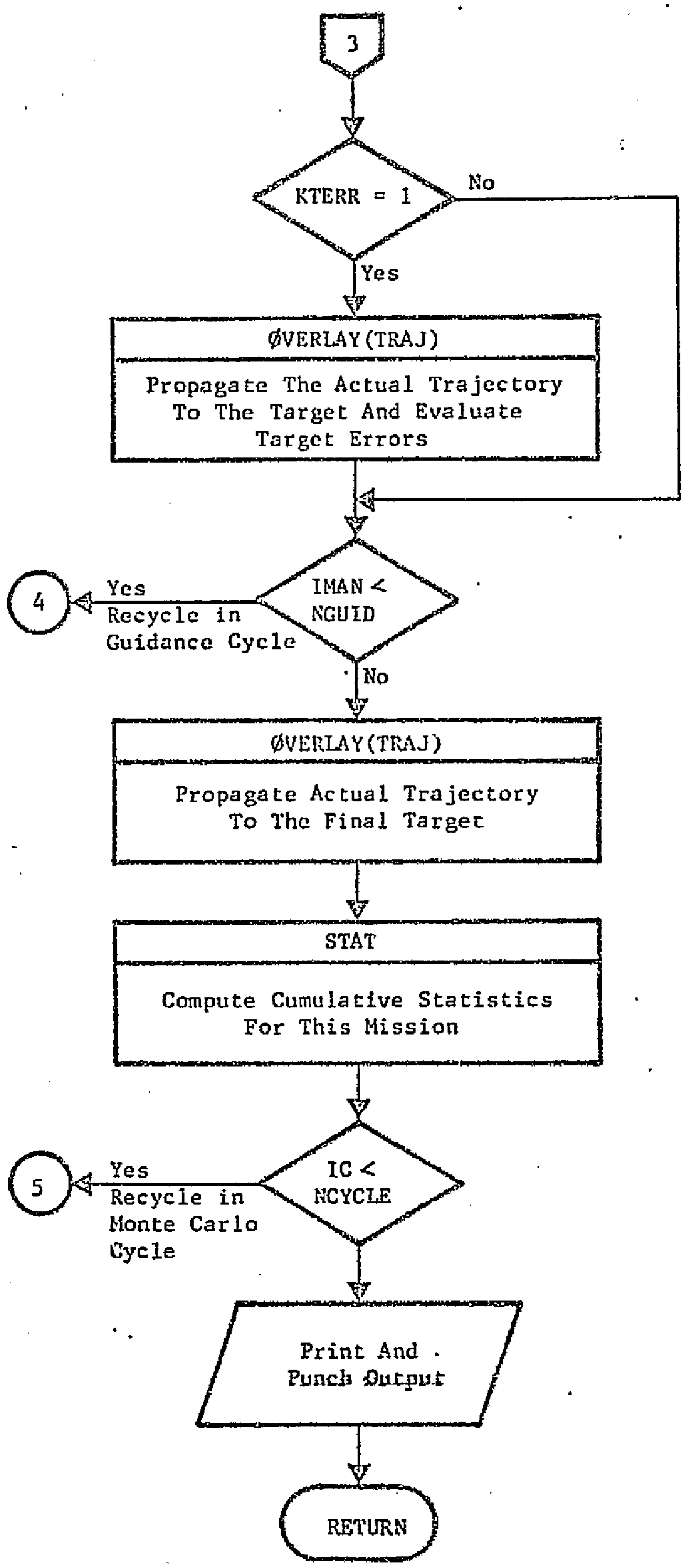
Common Blocks: CØNST, CYCLE, DYNØS, EDIT, EPHEM, IASTM, SIM1, ISIM1, SIM2, ISIM2, SIMLAB, STØREC, TIME, TRAJ1, TRAJ2, WORK, (BLANK)

Logic Flow:


SIMSEP
DATAS
Read Namelists And Initialize Program Data
SET(+4)
Initialize Actual, Estimated, And Reference Trajectory Constants And Parameters
I NREF = 1
Yes
No
REFTRJ
Compute reference trajectory conditions at the guidance and target points
4
Start of Monte Carlo Cycle
IC = IC+1
SET(-3)
Set All Nominal Working Arrays Used By TRAJ With The Reference Values
1


1
CSAMP
Sample PG To Form XA
EPHSMP
Sample Ephemeris Error To Form Actual Ephemeris Planet Orbits
ERRSMP
Sample S/C Parameter Errors To Form Actual THRUST, SCMASS, Etc.
SET(+1)
Save All Actual Parameters That Have Been Sampled In The SS1-Arrays
5
Start of Guidance Cycle
IMAN = IMAN + 1
SET(-1)
Set All Nominal Working Arrays With The Actual Parameter Values
OVERLAY(TRAJ)
Propagate Actual Trajectory To Guidance Event
ØD
Sample Augmented Knowledge Covariance To Form Estimated State Vector XE
2


2
|IGUID| = 2
No
Yes
NLGUID
Compute Nonlinear Guidance Correction, DELTAU
LGUID
Compute Linear Guidance Correction, DELTAU
ICNVEG≥1
No
Divergence
Yes
Divergence has Occurred In The Nonlinear Guidance Logic
4
ICNVEG ≥ 2
No
Weak Convergence
Yes, Convergence
EXGUID
Sample Guidance Event Execution Errors And Form Actual Correction
5
STAT
Compute Cumulative Statistics For This Guidance Event
EXGUID
Sample Guidance Event Execution Errors And Form Actual Correction
3


3
KTERR = 1
No
Yes
ØVERLAY(TRAJ)
Propagate The Actual Trajectory To The Target And Evaluate Target Errors
IMAN < NGUID
Yes
Recycle in Guidance Cycle
4
No
ØVERLAY(TRAJ)
Propagate Actual Trajectory To The Final Target
STAT
Compute Cumulative Statistics For This Mission
IC < NCYCLE
Yes
Recycle in Monte Carlo Cycle
5
Print And Punch Output
RETURN

3.4.1 Subroutine: CSAMP (EVEC, NN, REFVEC, SMPVEC, IRAN)

Purpose: To sample a n-dimensional covariance matrix in order to formulate a zero-mean, Gaussian, error vector which is added to the reference value.

Method: From an input array of eigenvalues corresponding to a specified covariance matrix in an uncorrelated representation, a standard Monte Carlo sampling technique is used to define a random vector. This random vector is then multiplied by the modal matrix of eigenvectors to rotate it back into the original state space. It is added to the reference vector to obtain a sample vector.

Remarks: This routine is used in SIMSEP for constructing random actual state vectors relative to the reference state at the initial time from the input control error covariance. It is also used to compute an augmented estimated state vectors from the input knowledge covariances at guidance events. The maximum dimension a covariance matrix may have is 20 X 20.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
EVEC	I	A	Variably dimensioned (NN X (NN+1)) array of eigenvectors and eigenvalues. The (NN X NN) square matrix

Variable	Input/ Output	Argument/ Common	Definition
			is the so-called modal matrix which has eigenvectors as columns. The (NN + 1) column vector is the (NN X 1) vector of eigenvalues.
NN	I	A	Dimensionality of the EVEC matrix.
REFVEC	I	A	Reference state vector to the sampled error vector is added.
SMPVEC	O	A	Sampled state vector which is different from REFVEC by the sampled error vector.
IRAN	I	A	Random number generator used.

Local Variables:

Variables	Definition
D	Sampled error vector to be added to REFVEC. Equivalences to elements in the WØRK common.

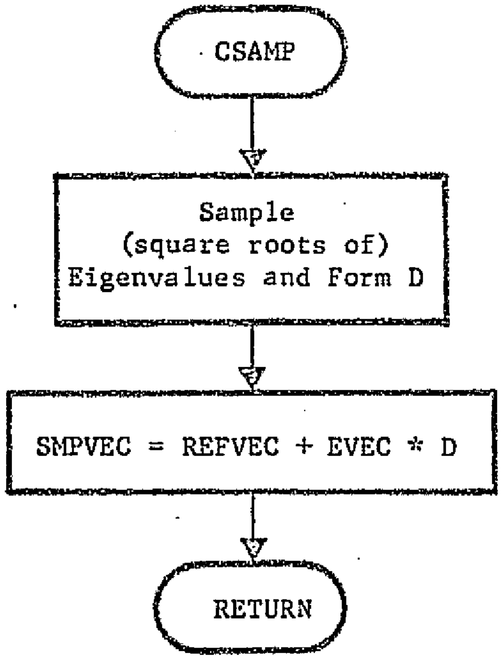
Subroutines Called: RNUM, MMAB, ADD

Calling Subroutines: SIMSEP, ØD, EPHSMP

Common Blocks: WØRK

Logic Flow:

<u>Logic Flow:</u>



3.4.2 Subroutine: DATAS

Purpose: To make calls to SDAT1 and SDAT2 in order to read the SIMSEP input.

Method: DATAS is a macro-logic routine which serves exclusively to call SDAT1 and SDAT2 in succession.

Input/Output: None

Local Variables: None

Subroutines Called: SDAT1, SDAT2

Calling Subroutines: SIMSEP

Common Blocks: None

Logic Flow: None

Pages 364 through 374 have been deleted.

3.4.4 Subroutine: EPHSMP (IPRNT)

Purpose: To make random samples from the input ephemeris planet error covariances and the gravitational constant uncertainties.

Method: A standard Monte Carlo sampling procedure is used to form discrete errors in the Cartesian state vector of the ephemeris planets. This sampling is made at a specified epoch and is transformed into changes in the Keplerian orbital elements. The analytic ephemeris is modified to reflect these ephemeris errors. Likewise, errors are computed for the solar and ephemeris planet gravitational constants.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
SMASS	I/O	C	Solar gravitational constant.
PMASS	I/O	C	Planetary gravitational constant.
PLANET	I	C	Hollerith array of planetary names.
CSAX	I/O	C	Analytic ephemeris semi-major axes.
CECC	I/O	C	Analytic ephemeris eccentricities.
CINC	I/O	C	Analytic ephemeris inclinations.

Variable	Input/ Output	Argument/ Common	Definition
CØMEG	I/O	C	Analytic ephemeris arguments of the ascending mode.
CØMEGT	I/O	C	Analytic ephemeris arguments of the apsis.
CMEAN	I/O	C	Analytic ephemeris mean anomalies and mean motions.
GMERR	I	C	One sigma uncertainties in the gravitational constants.
XEPH	I/O	C	Ephemeris planet state vector at epoch.
NEP2	I	C	Flag array specifying the ephemeris planets.
EPHERR	I	C	Eigenvector/eigenvalue representation of the ephemeris error covariance.
TEPH	I	C	Epoch at which the ephemeris errors are evaluated.

Local Variables:

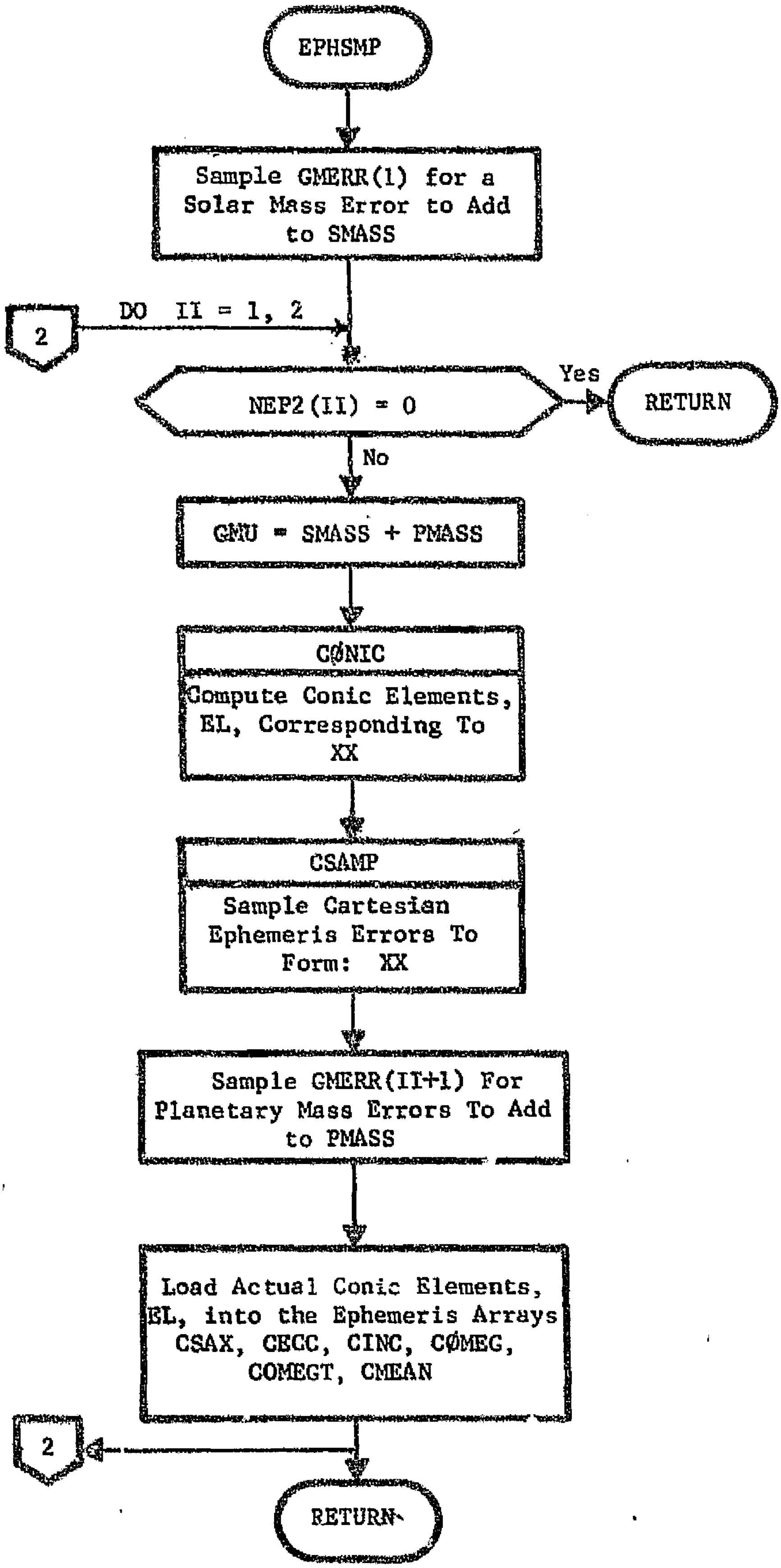
Variable	Definition
GMUS	Temporary storage for the solar gravitational constant.
GMU	Sum of sampled solar and planetary masses.
XX	Temporary storage for the sampled Cartesian ephemeris planet state.
EL	Temporary storage for the sampled orbital elements.

Subroutines Called: RNUM, CSAMP, CØNIC, CØPY, ZERØM

Calling Subroutines: SIMSEP

Common Blocks: CØNST, DYNØS, EPHEM, SIM1, ISIM1, WØRK

Logic Flow:


EPHSMP
Sample GMERR(1) for a Solar Mass Error to Add to SMASS
DO II = 1, 2
2
NEP2(II) = 0
Yes
RETURN
No
GMU = SMASS + PMASS
CØNIC
Compute Conic Elements, EL, Corresponding To XX
CSAMP
Sample Cartesian Ephemeris Errors To Form: XX
Sample GMERR(II+1) For Planetary Mass Errors To Add to PMASS
Load Actual Conic Elements, EL, into the Ephemeris Arrays CSAX, CECC, CINC, CØMEG, COMEGT, CMEAN
2
RETURN

Page 379 has been deleted.

3.4.5 Subroutine: ERRSMP

Purpose: To make random samples from input SEPS parameter errors, thrust biases and thrust process noise in order to formulate actual values for these parameters used during the propagation of an actual trajectory.

Methods: A standard Monte Carlo sampling procedure is used to compute random errors which are added to the reference values to form "actual" parameter values.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
SCMASS	I/O	C	Initial S/C mass.
ENGINE(10) (=EXHVEL)	I/O	C	Thrust exhaust velocity.
ENGINE(1) (=PØWERO)	I/O	C	Electric power at 1. A.U.
ENGINE(11) (=THREFF)	I/O	C	Thruster efficiency.
ENGINE(15) (=CRA)	I/O	C	Radiation pressure coefficient.
THRUST	I/O	C	Thrust control array.
TNØISE	O	C	Thrust control noise.
GTAU1	O	C	Thrust control noise time correlation coefficients for the first process.

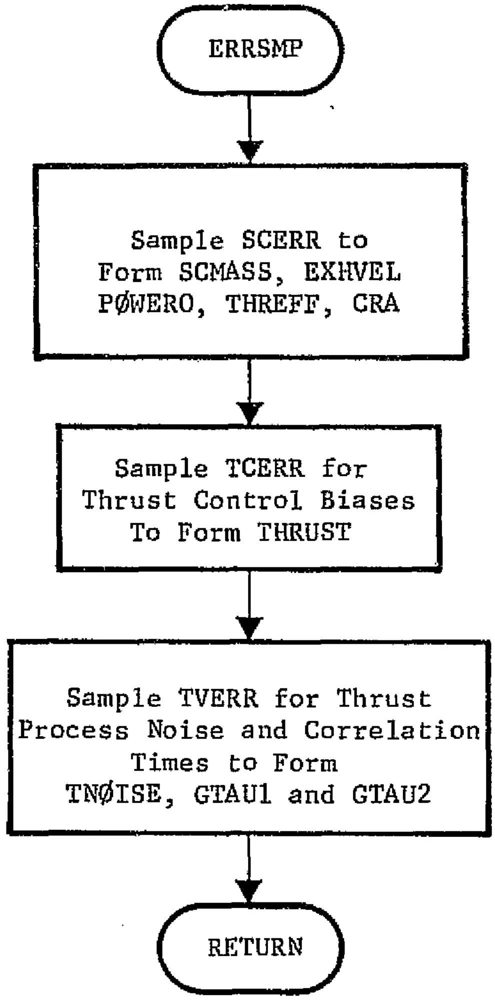
Variable	Input/ Output	Argument/ Common	Definition
GTAU2	O	C	Thrust control noise time correlation coefficients for the second process.
SCERR	I	C	SEPS parameter errors.
TCERR	I	C	Thrust control biases.
TVERR	I	C	Time varying thrust control errors.
JMAX	I	C	Total number of active thrust phases.
JMIN	I	C	Thrust phase number for the first active phase

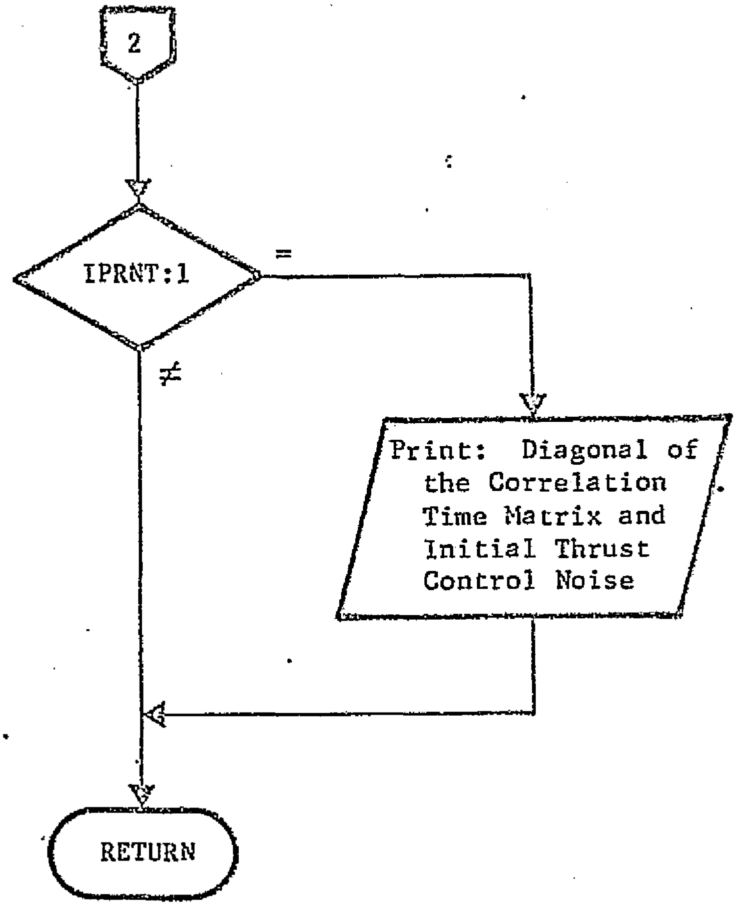
<u>Subroutines Called:</u> RNUM

<u>Calling Subroutines:</u> SIMSEP

<u>Common Blocks:</u> CØNST, DYNØS, SIM1, ISIM1, TIME, TRAJ1, TRAJ2, WØRK

Logic Flow:


ERRSMP
Sample SCERR to Form SCMASS, EXHVEL PØWERO, THREFF, CRA
Sample TCERR for Thrust Control Biases To Form THRUST
Sample TVERR for Thrust Process Noise and Correlation Times to Form TNØISE, GTAU1 and GTAU2
RETURN


2
IPRNT:1
=
≠
Print: Diagonal of the Correlation Time Matrix and Initial Thrust Control Noise
RETURN

3.4.6A Subroutine: EXGUID (XA, DELTAU, IMAN, IPRNT)

Purpose: To execute commanded thrust control changes or impulsive delta-velocity corrections which have been computed by the guidance algorithm.

Method: For a low thrust guidance event, the actual thrust controls are changed according to the commanded corrections computed by the guidance algorithm. These updated thrust controls still reflect thrust biases which were determined as random samples from the input error sources. For an impulsive guidance event, the commanded delta-velocity is corrupted by randomly sampled execution errors and is then added to the actual state vector as an instantaneous velocity change.

Input/Output:

Variables	Input/ Output	Argument/ Common	Definition
XA	I/O	A	Actual s/c state vector.
DELTAU	I	A	Commanded thrust control correction or delta-velocity change.
IMAN	I	A	Number of the current guidance event.
IPRNT	I	A	Print output flag.
EXVERR	I	C	Impulsive maneuver execution errors.
THRUST	I/O	C	Thrust control array.

Variable	Input/ Output	Argument/ Common	Definition
NTC	I	C	Number of active thrust controls.
IGL	I	C	Guidance law specification flag.

Local Variables:

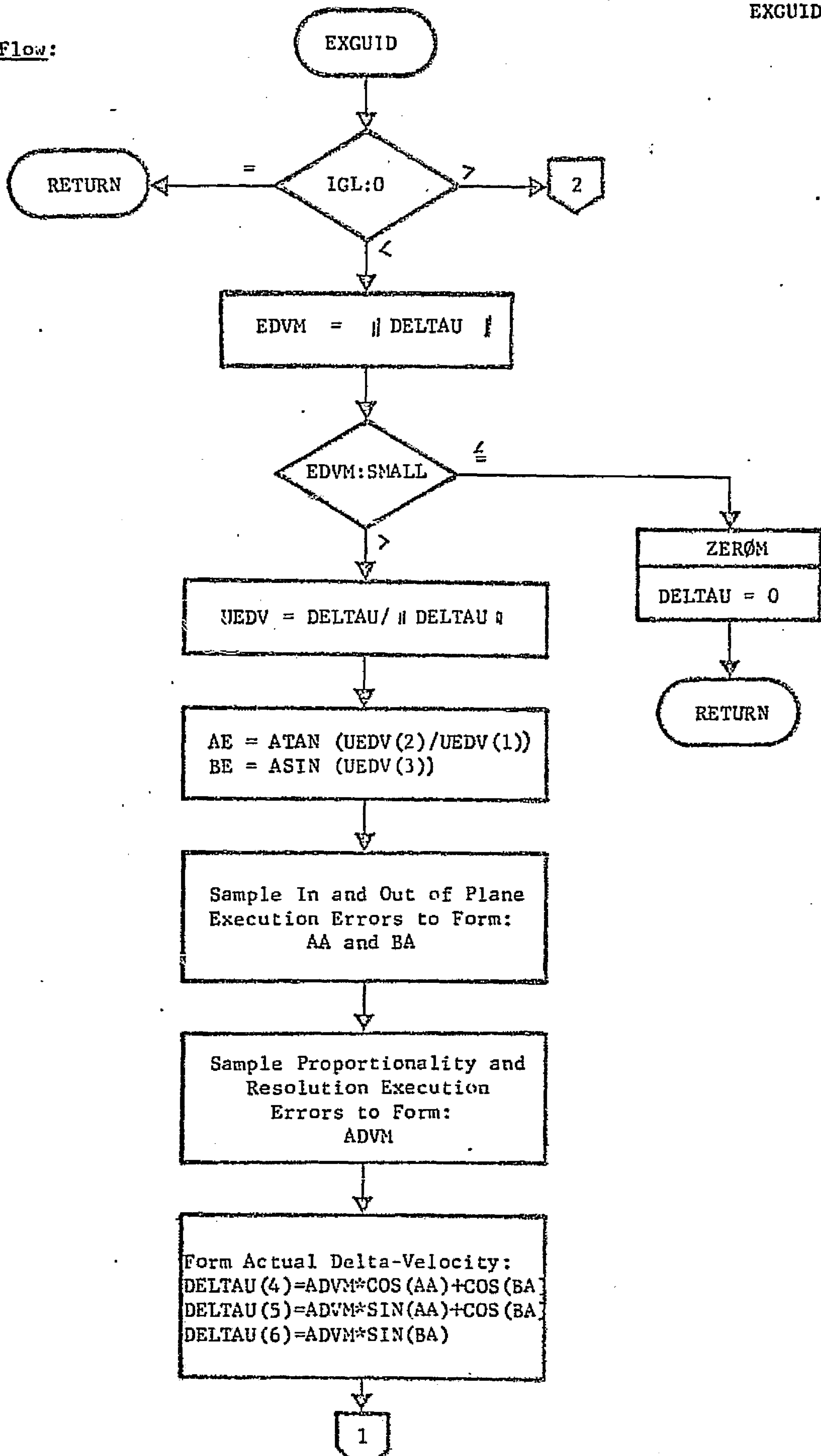
Variable	Definition
EDVM	Magnitude of the commanded delta-velocity correction.
ADVM	Magnitude of the actual delta-velocity correction.
UEDV	Unitized estimated delta-velocity vector.
AE	Angle measured in the ecliptic plane from the positive X-axis to the projection of the commanded delta-velocity correction.
BE	Angle measured out of the ecliptic plane to the commanded delta-velocity correction.
AA	Angle measured in the ecliptic plane from the positive X-axis to the projection of the actual delta-velocity correction.
BA	Angle measured out of the ecliptic plane to the actual delta-velocity correction.

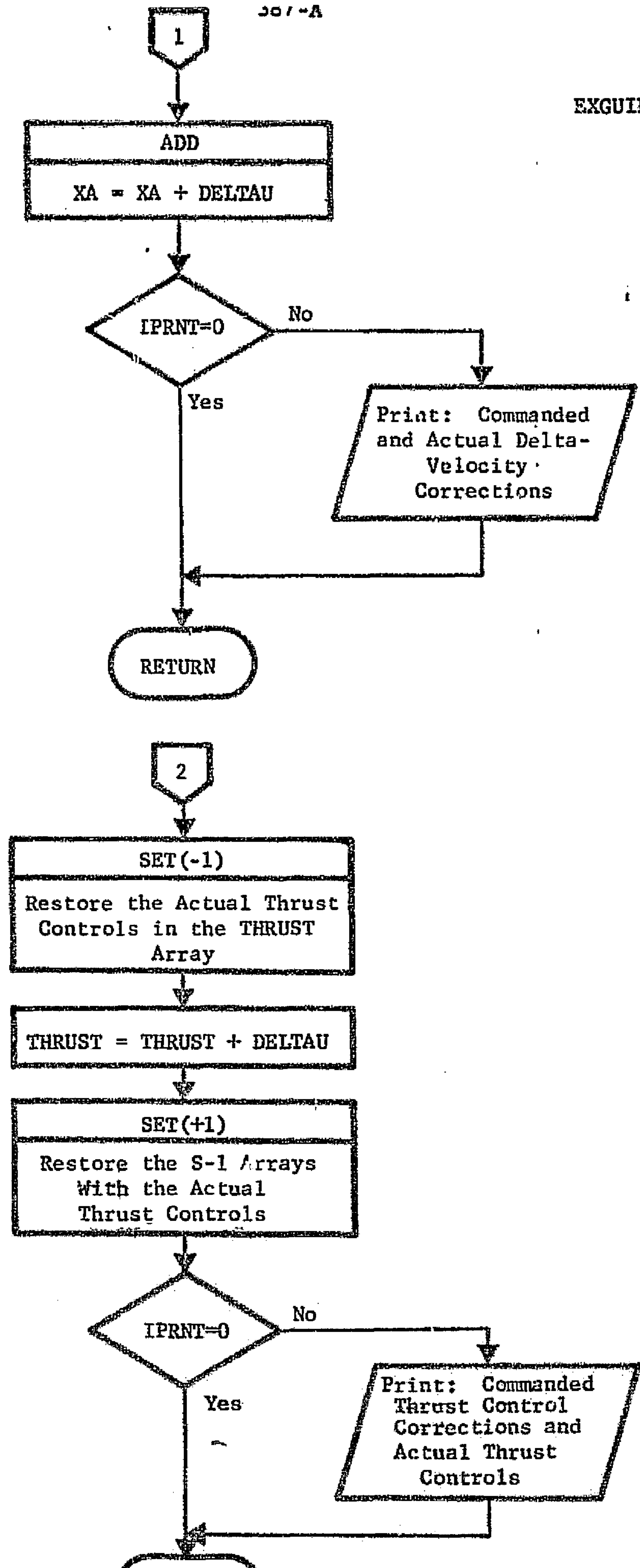
Subroutines Called: VECMAG, UNITV, RNUM, ZERØM, ADD, SET, MATØUT, CØPY

Calling Subroutines: SIMSEP

Common Blocks: CØNST, DYNØS, IASTM, SIM1, ISIM1, SIMLAB, STØREC, TRAJ1

Logic Flow:


EXGUID
RETURN
=
IGL:0
>
2
<
EDVM = ‖ DELTAU ‖
≤
EDVM:SMALL
>
ZERØM
DELTAU = 0
RETURN
UEDV = DELTAU/ ‖ DELTAU ‖
AE = ATAN (UEDV(2)/UEDV(1))
BE = ASIN (UEDV(3))
Sample In and Out of Plane Execution Errors to Form: AA and BA
Sample Proportionality and Resolution Execution Errors to Form: ADVM
Form Actual Delta-Velocity:
DELTAU(4)=ADVM*COS(AA)+COS(BA)
DELTAU(5)=ADVM*SIN(AA)+COS(BA)
DELTAU(6)=ADVM*SIN(BA)
1


1
ADD
XA = XA + DELTAU
IPRNT=0
No
Yes
Print: Commanded and Actual Delta-Velocity Corrections
RETURN
2
SET(-1)
Restore the Actual Thrust Controls in the THRUST Array
THRUST = THRUST + DELTAU
SET(+1)
Restore the S-1 Arrays With the Actual Thrust Controls
IPRNT=0
No
Yes
Print: Commanded Thrust Control Corrections and Actual Thrust Controls

3.4.6B **Subroutine:** GUIDMX (PHI, THETA, ETA, GAMMA, NC, NT, IGUID, IMAN, CØNWT)

Purpose: To calculate the guidance matrix used by the linear guidance algorithm.

Method: The guidance matrix, Γ , is computed from trajectory sensitivities evaluated about the reference trajectory according to the guidance policy specified during input. The computational steps in formulating Γ are discussed in the Analytic Manual, Section 7.3.1. Once the guidance matrix has been determined, it is stored and used on successive Monte Carlo cycles, thus eliminating the need to re-evaluate trajectory sensitivities.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
PHI	I	A	State to state transition matrix, Φ .
THETA	I	A	Control variable to state component transition matrix, θ_u.
ETA	I	A	State to target variable transformation matrix, η.
GAMMA	O	A	Guidance matrix, Γ .
NC	I	A	Number of control variables.
NT	I	A	Number of target variables.
IGUID	I	A	Guidance maneuver type flag.

Variable	Input/ Output	Argument/ Common	Definition
IMAN	I	A	Guidance event number.
CØNWT	I	A	Weighting factors for the control variables.

Local Variables:

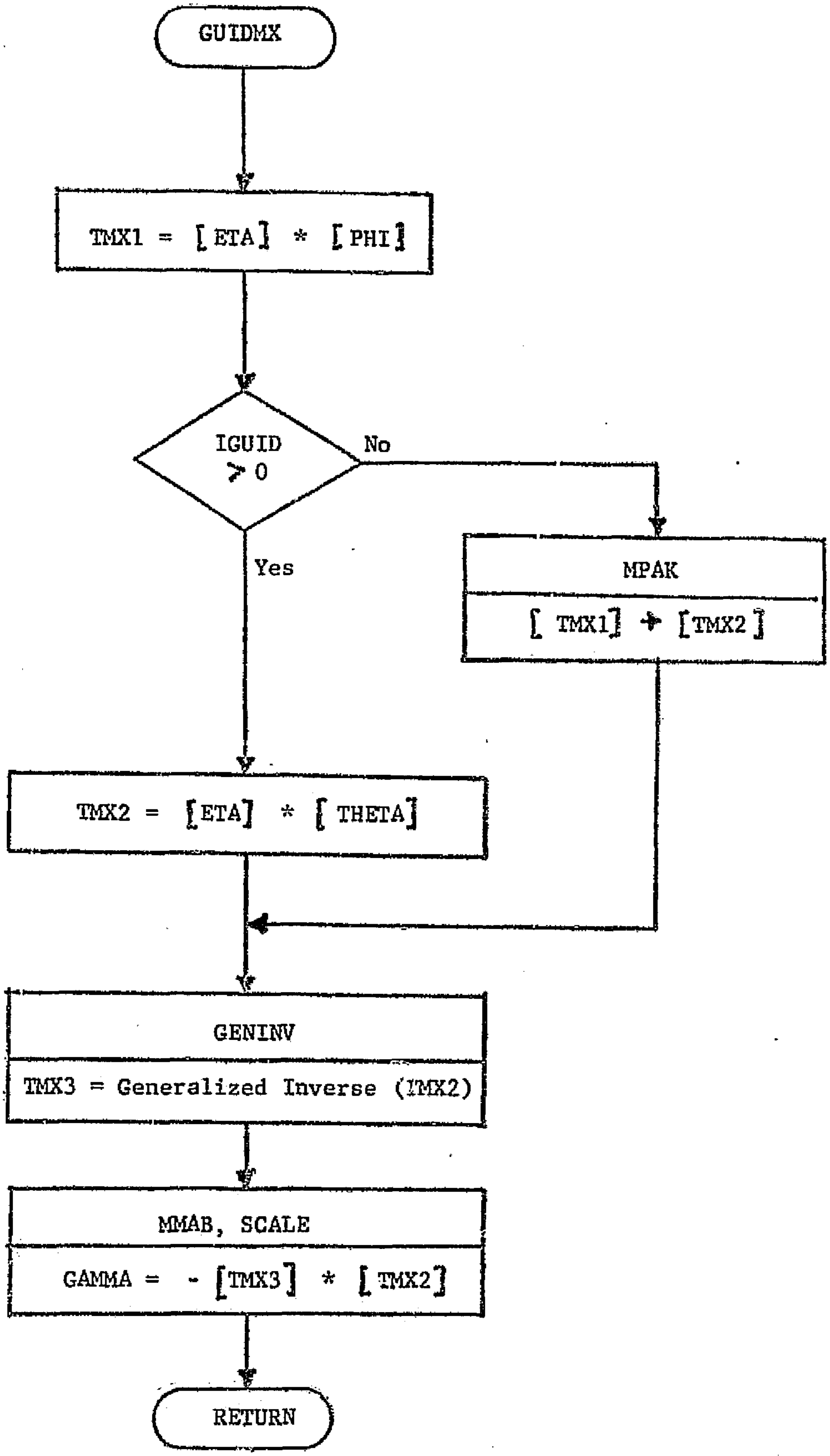
Variable	Definition
TMX1 TMX2 TMX3	Temporary matrices storing intermediate calculations.

Subroutines Called: GENINV, MMAB, MPAK, SCALE

Calling Subroutine: REFTRJ

Common Blocks: None

Logic Flow:


GUIDMX
TMX1 = [ETA] * [PHI]
IGUID
> 0
No
Yes
MPAK
[TMX1] + [TMX2]
TMX2 = [ETA] * [THETA]
GENINV
TMX3 = Generalized Inverse (TMX2)
MMAB, SCALE
GAMMA = - [TMX3] * [TMX2]
RETURN

3.4.7 Subroutine: LGUID (XE, IMAN, IPRNT, DELTAU)

Purpose: To compute low thrust or impulsive guidance corrections using a linear, non-iterative guidance law.

Method: Using the linear guidance matrix, Γ, formulated in GUIDMX, LGUID computes a set of low thrust or impulsive corrections according to the matrix equation

$$A \underline{u} = \Gamma \, \delta \underline{X}_E ,$$

where $\delta \underline{X}_E$ is the state vector difference between the estimated and reference trajectory state at the guidance point.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
XE	I	A	Estimated S/C state vector.
IMAN	I	A	Number of the current guidance event.
IPRNT	I	A	Print output flag.
DELTAU	O	A	Output vector of low thrust or impulsive velocity corrections.
SMAT	I	C	Saved guidance matrix previously computed.
NTC	I	C	Number of control variables.
RXGE	I	C	Reference trajectory state vector at the guidance point.

Local Variables:

Variable	Definition
DXE	Deviation of the estimated state vector relative to the reference trajectory at the guidance point.
GAMMA	Guidance matrix, Γ .
EDU	Temporary storage for the computed control correction.

Subroutines Called: CØPY, MMAB, SUB

Calling Subroutines: SIMSEP

Common Blocks: IASTM, SIM1, ISIM1, SIMLAB, STØREC, TIME, WØRK

Logic Flow: None

3.4.8 Subroutine: NLGUID (XE, IMAN, IPRNT, DELTAU, ICNVEG)

Purpose: To compute low thrust or impulsive guidance corrections using a nonlinear guidance algorithm.

Method: The estimated state is propagated to the designated target time where target errors relative to the reference target conditions are evaluated. State variations with respect to guidance controls are computed with the estimated trajectory propagation. From the target errors and the resultant sensitivity matrix, a linear control correction is calculated and applied as an update to the current controls. This process is repeated until the target errors are within specified tolerances. If the target tolerances are not satisfied after NMAX iterations, further guidance corrections for the current Monte Carlo mission are aborted and the mission is ended. A more complete discussion of the nonlinear guidance problem and the method of solution which has been implemented here is given in the Analytic Manual, Section 7.3.4.

INPUT/OUTPUT: VARIABLE	INPUT/ OUTPUT	ARGUMENT/ COMMON	DEFINITION
XE	I	A	Estimated S/C state vector.
IMAN	I	A	Number of the current guidance event.
IPRNT	I	A	Print output flag.
DELTAU	0	A	Computed low thrust or impulsive control corrections.
ICNVEG	0	A	Convergence flag. = 0, No convergence after ITMX iterations or after the quadratic error function, Q, has increased on three successive iterations. = 1, Weak convergence after ITMX iterations and Q being less than AOK. = 2, Strong convergence (Q ≤ 1).
TØL	I	C	Array of target error tolerances used in computing the quadratic error function.

VARIABLE	INPUT/ OUTPUT	ARGUMENT COMMON	DEFINITION
IGL	I	C	Flag designating the type of guidance correction to be computed. If IGL = +2, the guidance is low thrust. If IGL = -2, the guidance is impulsive.
ITMX	I	C	Maximum number of guidance iterations allowed. (Input as NMAX).
AØK	I	C	Weak convergence tolerance.
IEPH	I	C	Code for the ephemeris body.
NTPL	I	C	Code for the target body.
NTAR	I	C	Number of target variables.
NTC	I	C	Number of control variables.
TGE	I	C	Time of the guidance event.
TTAR	I	C	Designated target time.
LSTAR	I	C	List of target variable codes.
XTARG	I	C	Reference trajectory target conditions at the designated target time.
SMAT	I	C	Stored sensitivity matrix.
CØNWT	I	C	Control variable weights.
THRUST	I	C	Array of thrust controls.
STHRT2	I	C	Stored array of estimated thrust controls.
RXTAR	I	C	Reference trajectory state at the designated target time.
UNTAR	I	C	Conversion factor which convert target variables from internal to external units.
DVMXN	I	C	Maximum delta-velocity magnitude change.

VARIABLE	INPUT/ OUTPUT	ARGUMENT COMMON	DEFINITION
LJH	I	C	Array of indices which identify the position in the THRUST array of the active controls.
PHI	O	C	State to state transition matrix between TGE and TTARG.
THETA	O	C	Controls to state transition matrix between TGE and TTARG.

LOCAL VARIABLES:

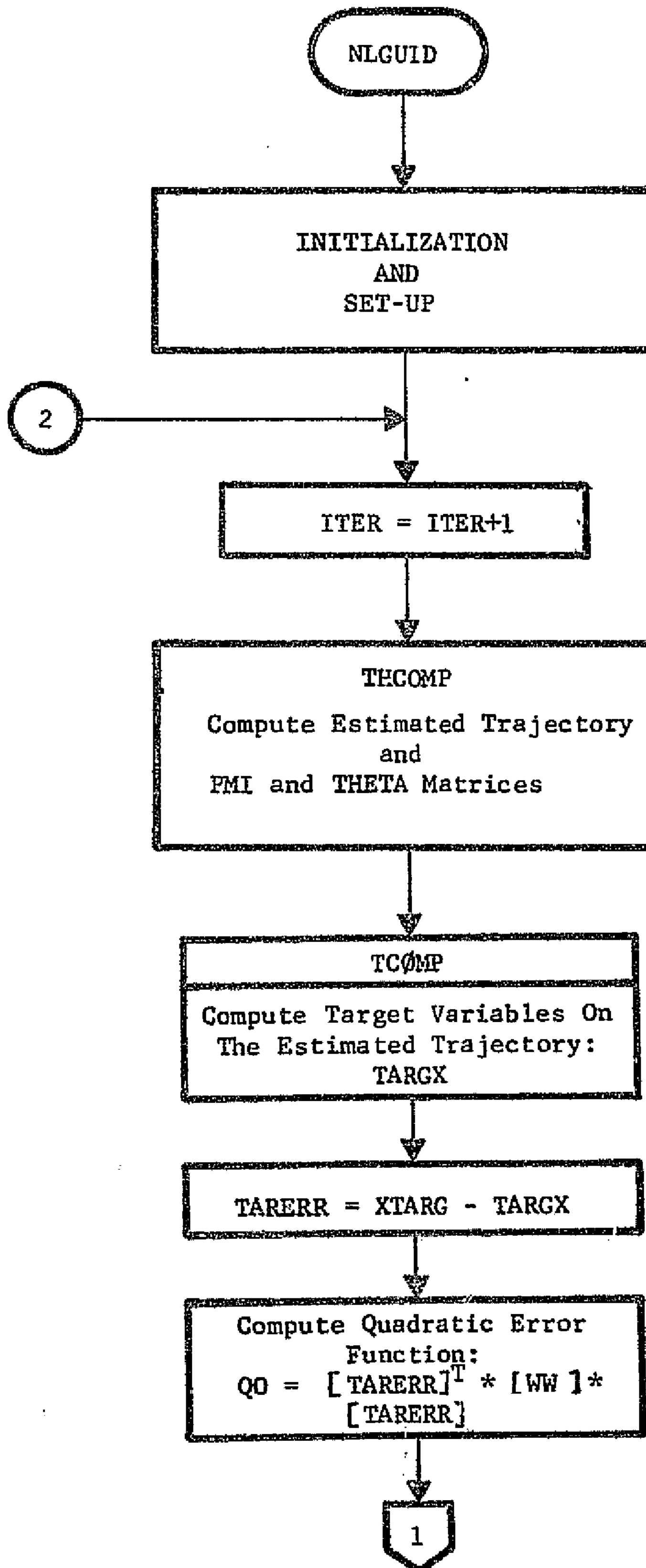
VARIABLE	DEFINITION
WW	Weighting matrix used in formulating the quadratic error function. WW is diagonal with the reciprocal target tolerances squared for the non-zero entries.
XXE	Estimated trajectory state vector at TSTØP.
ITER	Current iteration counter.
Q2	Value of the quadratic error function evaluated on two previous iterations.
Q1	Value of the quadratic error function evaluated on one previous iteration.
Q0	Current value of the quadratic error function.
ETA	Transformation matrix mapping differential state variables into differential target variables.
EDV	Delta-velocity guidance correction at the current iteration.
EDU	Delta-thrust-control guidance correction at the current iteration.
TARGX	Target variables evaluated on the estimated trajectory at TSTØP.
TARERR	Target error at TSTØP.
GAMMA	Guidance matrix which maps target errors into control variables.

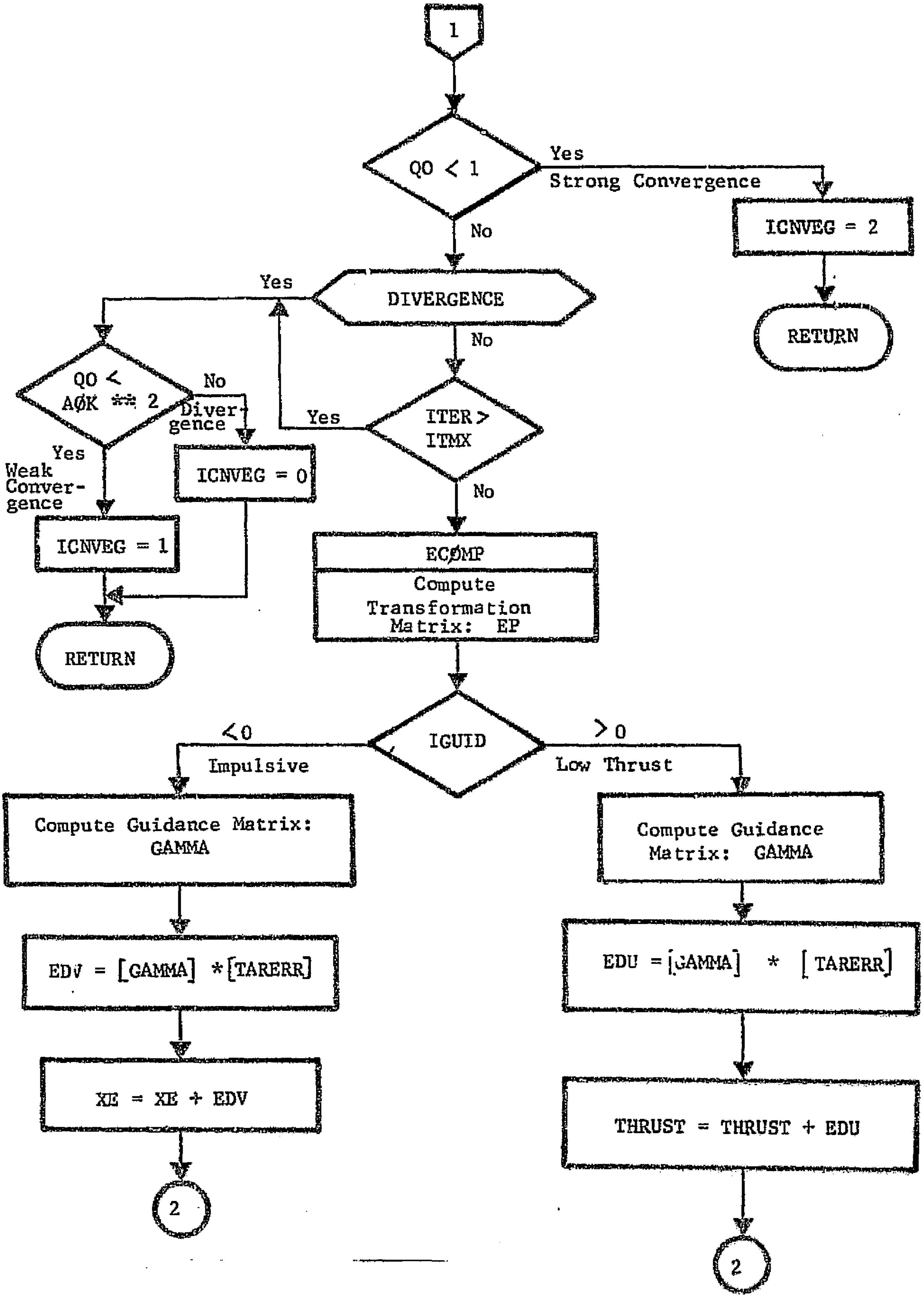
Subroutines Called: ZERØM, MATØUT, CØPY, SET, MMAB, GENINV, VECMAG, SCALE, ADD, THCØMP, TCØMP, SUB, MMATBA, ECØMP,

Calling Subroutines: SIMSEP

Common Blocks: CØNST, TRAJ1, TRAJ2, SIM1, ISIM1, TIME, (BLANK)

Logic Flow:


NLGUID
INITIALIZATION
AND
SET-UP
2
ITER = ITER+1
THCOMP
Compute Estimated Trajectory
and
PMI and THETA Matrices
TCØMP
Compute Target Variables On
The Estimated Trajectory:
TARGX
TARERR = XTARG - TARGX
Compute Quadratic Error
Function:
QO = [TARERR]T * [WW]*
[TARERR]
1


1
QO < 1
Yes
Strong Convergence
ICNVEG = 2
RETURN
No
DIVERGENCE
Yes
No
QO <
AØK ** 2
No
Diver-
gence
ICNVEG = 0
Yes
Weak
Conver-
gence
ICNVEG = 1
RETURN
ITER >
ITMX
Yes
No
ECØMP
Compute
Transformation
Matrix: EP
IGUID
<0
Impulsive
>0
Low Thrust
Compute Guidance Matrix:
GAMMA
EDV = [GAMMA] *[TARERR]
XE = XE + EDV
2
Compute Guidance
Matrix: GAMMA
EDU = [GAMMA] * [TARERR]
THRUST = THRUST + EDU
2

Pages 396 through 402 have been

deleted

3.4.9A **Subroutine:** ØD (XA, XE, IMAN, IPRNT)

Purpose: To estimate the s/c state vector and parameters which have been augmented to the state at a guidance event.

Method: Since an explicit orbit determination process and measurement models are not included in SIMSEP, ØD, in effect, performs the state estimation function. A knowledge covariance, which has been transformed into an eigenvector/eigenvalue representation, is randomly sampled to form an error, δX_E, in the estimated state vector relative to the actual, i.e., $\delta X_E = X_E - X_A$. If parameters such as ephemeris errors, thrust biases, etc., have been augmented to the six-component Cartesian state, estimated errors for these parameters are simultaneously computed by sampling an augmented knowledge covariance. The formulated error vector is added to the corresponding actual values to define an estimated state and estimates of the augmentation parameters to be used in calculating guidance corrections.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
XA	I	A	Actual s/c state vector (position and velocity).

Variable	Input/ Output	Argument/ Common	Definition
XE	O	A	Estimated s/c state vector (position and velocity).
IMAN	I	A	Number of the current guidance event.
IPRNT	I	A	Print output flag.
BLANK	I	C	Array of eigenvector and eigenvalues corresponding to the augmented knowledge covariance.
ENGINE(1) (= PØWERO)	O	C	Estimated electric power at 1 A.U.
SPO3	I	C	Saved reference value of the electric power at 1 A.U.
ENGINE(10) (= EXHVEL)	O	C	Estimated thrust exhaust velocity.
SEXV3	I	C	Saved reference value of the thrust exhaust velocity.
ENGINE(11) (= THREFF)	O	C	Estimated thruster efficiency.
STEFF3	I	C	Saved reference value of the thruster efficiency.
ENGINE(15) (= CRA)	O	C	Estimated radiation pressure coefficient.
SCRA3	I	C	Saved reference value of the radiation pressure.
SCMASS	O	C	Estimated SEPS mass.
RMGE	I	C	Reference SEPS mass.
THRUST	O	C	Estimated thrust control array.
STHRT3	I	C	Saved reference thrust control array.

Variable	Input/ Output	Argument/ Common	Definition
KDIM	I	C	Dimension of the augmented knowledge covariance.
IEPH	I	C	Ephemeris planet number for the current knowledge covariance.
NEP2	I	C	Active actual ephemeris planet number.
SMASS	I/O	C	Estimated solar gravitational constant.
PMASS	I/O	C	Estimated planetary gravitational constant.
XEPH	I/O	C	Estimated ephemeris planet Cartesian state vector.
GMERR	I	C	Solar and planetary gravitational constant uncertainties.
CSAX	O	C	Estimated semi-major axis for the ephemeris planet.
CECC	O	C	Estimated eccentricity for the ephemeris planet.
CINC	O	C	Estimated inclination for the ephemeris planet.
CØMEG	O	C	Estimated longitude of the ascending node for the ephemeris planet.
CØMEGT	O	C	Estimated longitude of periapsis for the ephemeris planet.
CMEAN	O	C	Estimated mean anomaly for the ephemeris planet.

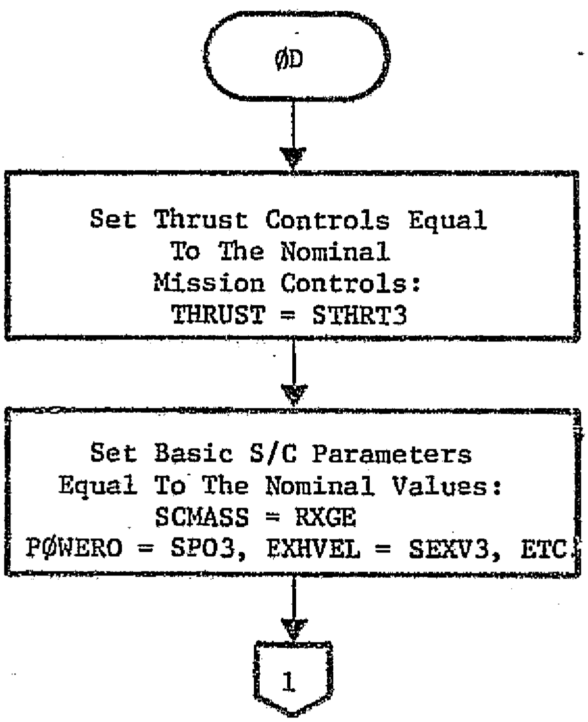
Local Variables:

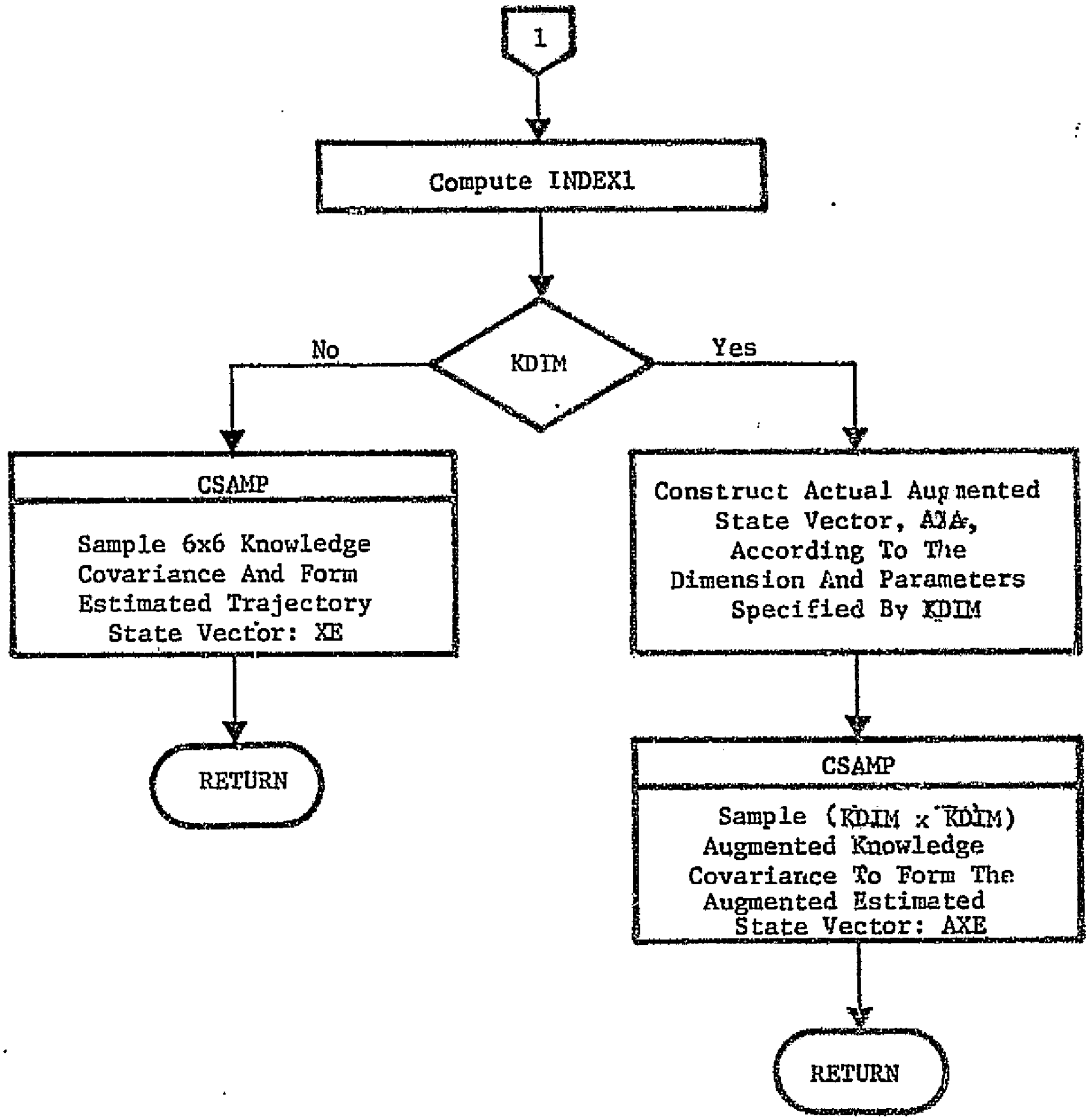
Variable	Definition
AXA	Augmented actual state vector. The dimension and packing are determined by KTY.
AXE	Augmented estimated state vector. Like AXA, the dimension and packing are determined by KTY.
EL	Estimated osculating ephemeris planet orbital elements.
INDEX1	Index identifying the position in the EVEC matrix of the first element corresponding to the current augmented knowledge covariance.

Subroutines Called: ZERØM, CSAMP, CØPY, CØNIC

Calling Subroutines: SIMSEP

Common Blocks: CØNST, TRAJ1, EPHEM, TIME, SIM1, STØREC, WØRK, ISIM1 (BLANK)




1
Compute INDEX1
KDIM
No
Yes
CSAMP
Sample 6x6 Knowledge Covariance And Form Estimated Trajectory State Vector: XE
RETURN
Construct Actual Augmented State Vector, AXA, According To The Dimension And Parameters Specified By KDIM
CSAMP
Sample (KDIM x KDIM) Augmented Knowledge Covariance To Form The Augmented Estimated State Vector: AXE
RETURN

3.4.9B Subroutine: ØPSTAT

Purpose: To output statistics evaluated during the Monte Carlo mission simulations.

Method: After completion of Monte Carlo cycles in SIMSEP, ØPSTAT transforms variances and covariances which characterize the statistics of the "real world" trajectories into standard deviations and correlation coefficients. The standard deviations, correlations, and means are printed as a part of the standard SIMSEP output whenever the number of Monte Carlo cycles is greater than one. Arrays of these numbers are also punched (if requested by the user) in a format ready to initialize a subsequent SIMSEP run.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
NGUID	I	C	Number of guidance events occurring on the mission.
NSAMP(i)	I	C	Number of Monte Carlo cycles executed in accumulating statistics for i^{th} guidance events.
GCCØV(i)	I	C	Control error covariance and vector mean evaluated at the i^{th} guidance event.
GMCØV(i)	I	C	S/C mass variance and mean evaluated at the i^{th} guidance event.

Variable	Input/ Output	Argument/ Common	Definition
DVCØV(i)	I	C	Delta-velocity covariance and vector mean evaluated for impulsive maneuvers at the i^{th} guidance event.
DVMAGS(i)	I	C	Delta-velocity magnitude variance and mean for impulsive maneuvers at the i^{th} guidance event.
CNCØV(i)	I	C	Thrust control correction covariance and means evaluated for low thrust maneuvers at the i^{th} guidance event.
NTC(i)	I	C	Number of low thrust controls active for the i^{th} guidance event.
TCCØV(i)	I	C	Control error covariance and vector mean evaluated at the target time on the i^{th} guidance event.
TMCØV(i)	I	C	S/C mass variance and mean evaluated at the target time on the i^{th} guidance event.
TERCØV(i)	I	C	Target error covariance and means evaluated at the target time on the i^{th} guidance event.
NTAR(i)	I	C	Number of target variable for the i^{th} guidance event.
MC(i)	I	C	Number of Monte Carlo cycles executed in accumulating statistics.
ENDCØV	I	C	Control error covariance and vector mean evaluated at the trajectory end time (TEND).
AMASS	I	C	S/C mass variance and mean evaluated at the trajectory end (TEND).

Variable	Input/ Output	Argument/ Common	Definition
ADVT	I	C	Delta-velocity magnitude variance and mean evaluated for all impulsive maneuvers.
ATHCØV	I	C	Covariance of active thrust controls used throughout the mission for all low thrust maneuvers executed.
KATHC	I	C	Dimension of the ATHCOV matrix.

Local Variables: None

Subroutines Called: MATØUT, SYMUP, VARSD

Calling Subroutines: SIMSEP

Common Blocks: SIM1, ISIM1, SIM2, ISIM2

Logic Flow: None

3.4.90 Subroutine: REFTRJ

Purpose: (1) To compute reference trajectory conditions, e.g., state, mass, sensitivities, etc., at the guidance points; (2) to evaluate reference trajectory target conditions at designated target times; and (3) to compute the guidance matrix to be used at linear guidance events.

Method: REFTRJ performs the trajectory calculations necessary whenever INREF is read as zero during the $SIMSEP namelist input. These calculations are done by repetitively calling either the TRAJ overlay or the THCØMP subroutine. In addition, REFTRJ prints and punches the reference trajectory data so that they may be used to initialize subsequent SIMSEP runs (with INREF = 1).

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
TGE	I	C	Epoch of a guidance event.
TTAR	I	C	Designated target epoch.
NGUID	I	C	Number of guidance events.
NTAR	I	C	Number of target variables.
NTC	I	C	Number of controls.
NTPL	I	C	Target planet code.
IGL	I	C	Guidance law flag.
LSTAR	I	C	List of target variable codes.

Variable	Input/ Output	Argument/ Common	Definition
RXGE	O	C	Reference trajectory state at the guidance event.
RMGE	O	C	Reference S/C mass at the guidance event.
RXTAR	O	C	Reference trajectory state at the target time.
RMTAR	O	C	Reference S/C mass at the target time.
XTARG	O	C	Reference target conditions at the target time.
XEND	O	C	Reference trajectory state at the final trajectory time (TEND).
MEND	O	C	Reference S/C mass at the final trajectory time.
SMAT	O	C	Sensitivity or guidance matrix for guidance maneuvers.
PHI	O	C	State to state transition matrix.
THETA	O	C	Thrust controls to state transition matrix.

Local Variables:

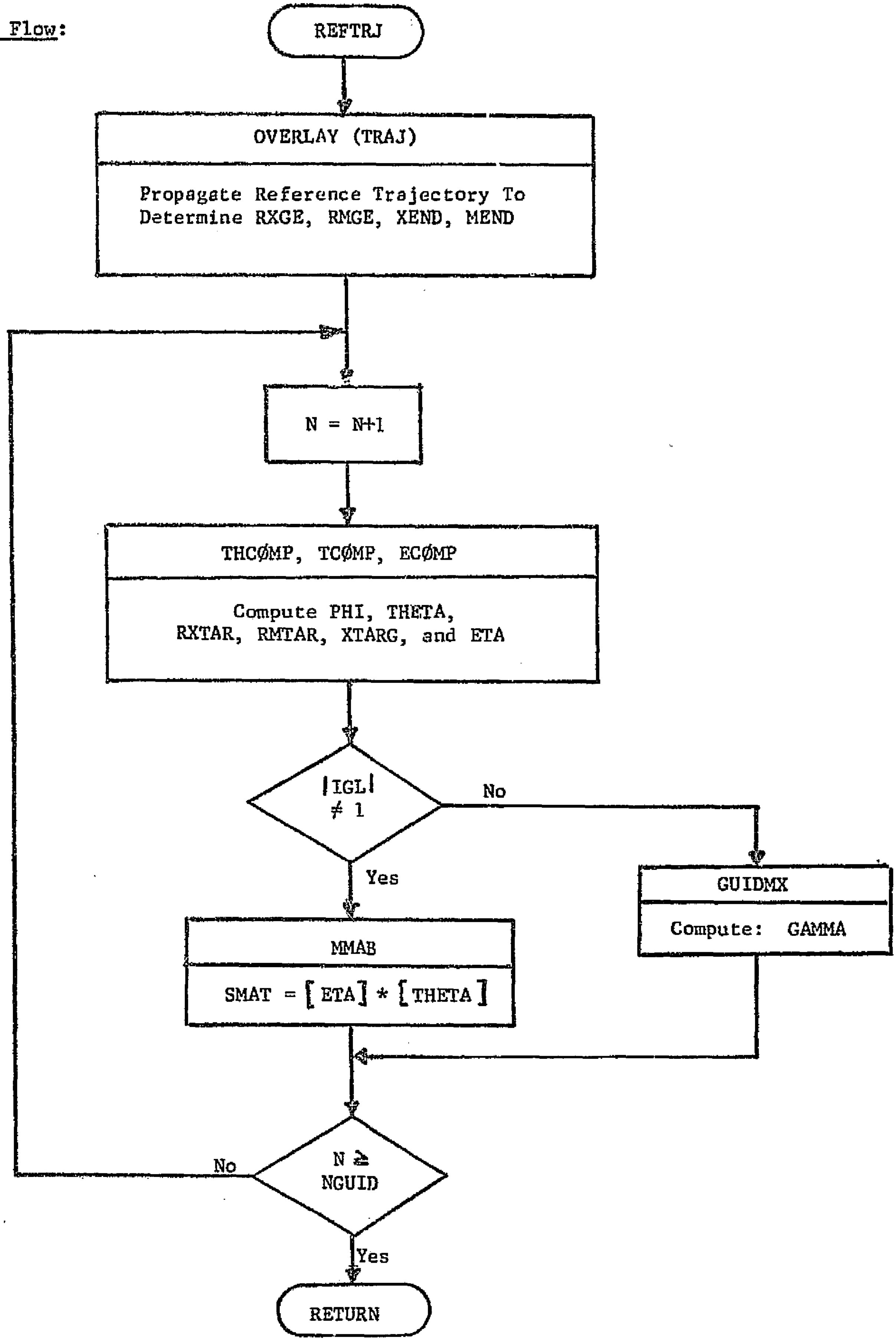
Variable	Definition
ETA	State to target variable transformation matrix.
GAMMA	Linear guidance matrix.
TMX1	Temporary storage of intermediary calculations.

Subroutines Called: CØPY, ECØMP, GUIDMX, MMAB, MPAK, TRAJ, TCØMP, THCØMP

Calling Subroutines: SIMSEP

Common Blocks: CØNST, EPHEM, IASTM, SIM1, ISIM1, SIMLAB, TIME, TRAJ1, TRAJ2, (BLANK)

Logic Flow:


REFTRJ
OVERLAY (TRAJ)
Propagate Reference Trajectory To Determine RXGE, RMGE, XEND, MEND
N = N+1
THCØMP, TCØMP, ECØMP
Compute PHI, THETA, RXTAR, RMTAR, XTARG, and ETA
|IGL| ≠ 1
No
Yes
GUIDMX
Compute: GAMMA
MMAB
SMAT = [ETA] * [THETA]
N ≥ NGUID
No
Yes
RETURN

3.4.9D Subroutine: SDAT1

Purpose: To read input data from the $SIMSEP namelist and to initialize the trajectory simulation mode.

Method: Once the default values have been initialized, the $SIMSEP namelist is read from input. Names, dimensions, and definitions for variables contained in $SIMSEP are discussed in the User's Manual (Section 2.4, page 37). The input data are processed and stored in common blocks so that they may be used by Monte Carlo cycle logic in SIMSEP. Variables contained in this namelist control the degree of data preparation and computational operations performed within the main cycle of the program.

Remarks: Many of the variables appearing in SDAT1 are initialized from namelist with units specified in the User's Manual. Before they are transmitted to other routines and used by the program, they are converted to internal units which are kg, kw, km, sec, km/sec, and radians.

Input/Output:

Variable	Input/ Output	Namelist/ Common	Definition
AØK	I/0	N/C	Backup convergence tolerance for weak convergence test.
CPMA	I/0	N/C	Computer processing time limit for the current SIMSEP run.
DVMXN	I/0	N/C	Maximum delta-velocity magnitude step.
INREF	I/0	N/C	State vector and trajectory parameter read-in flag.
IØUT	I/0	N/C	Print output flag.
IPUNCH	I/0	N/C	Punch output flag.
IRAN	I/0	N/C	Random number seed.
NCYCLE	I/0	N/C	Number of Monte Carlo cycles to be run.

Input/Output:

Variable	Input/ Output	Namelist/ Common	Definition
NGUID	I/O	N/C	Number of guidance events to be executed on each Monte Carlo mission simulation.
EPHERR	I/O	N/C	Ephemeris error matrices for the ephemeris bodies.
GMERR	I/O	N/C	Gravitational constant errors.
NEP2	I/O	N/C	Active ephemeris planet numbers.
XEPH	O	C	Cartesian state of the ephemeris body at TEPH.
PG	I/O	N/C	S/C control error matrix.
TEPH	I/O	N/C	Epoch of evaluation of the ephemeris errors.
EXVERR	I/O	N/C	Midcourse velocity correction execution errors.
SCERR	I/O	N/C	SEP and S/C errors.
TCERR	I/O	N/C	Thrust bias errors.
TVERR	I/O	N/C	Thrust process noise.
ADVT	I/O	N/C	Total delta-velocity magnitude statistics.
ENDCØV	I/O	N/C	Accumulated S/C control error statistics at TEND.
AMASS	I/O	N/C	Accumulated S/C mass statistics at TEND.
ATHCØV	I/O	N/C	Accumulated total thrust control statistics.
XEND	I/O	N/C	Reference trajectory state vector at TEND.

Input/Output:

Variable	Input/ Output	Namelist/ Common	Definition
MEND	I/O	N/C	Reference S/C mass at TEND.
SPFIMP	I/O	N/C	Chemical propulsion system specific impulse.
DVMDØT	I/O	N/C	Chemical propulsion system mass flow rate.
MC	I/O	N/C	Number of previous Monte Carlo cycles.
KATHC	I/O	N/C	Dimension of the ATHCØV matrix.
JMAX	O	C	Number of the last active thrust control phase between trajectory times TSTART and TEND.
JMIN	O	C	Number of the first active thrust control phase after TSTART.

Local Variables: None

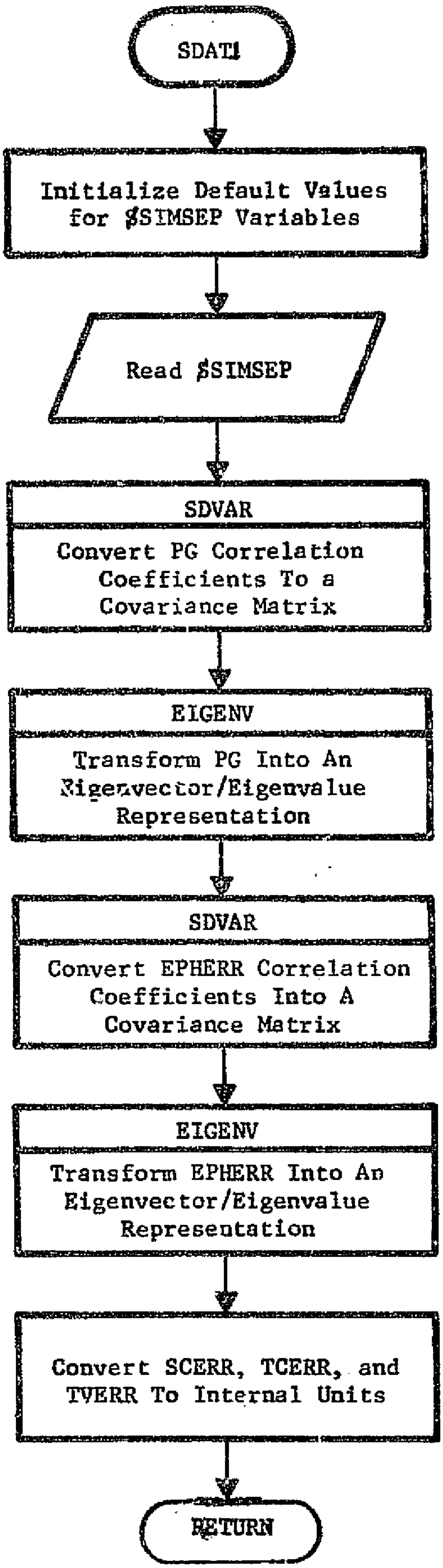
Subroutines Called: CØPY, EIGENV, EPHEM, MATØUT, SDVAR, ZERØM.

Calling Subroutines: DATAS

Common Blocks: CØNST, CYCLE, DYNØS, EDIT, EPHEM, SIM1, ISIM1, SIM2, ISIM2, SIMLAB, TIME, TRAJ1, TRAJ2.

Logic Flow:

Logic Flow:


SDAT1
Initialize Default Values for $SIMSEP Variables
Read $SIMSEP
SDVAR
Convert PG Correlation Coefficients To a Covariance Matrix
EIGENV
Transform PG Into An Eigenvector/Eigenvalue Representation
SDVAR
Convert EPHERR Correlation Coefficients Into A Covariance Matrix
EIGENV
Transform EPHERR Into An Eigenvector/Eigenvalue Representation
Convert SCERR, TCERR, and TVERR To Internal Units
RETURN

3.4.9E **Subroutine:** SDAT2

Purpose: To read input data from the $GUID namelist and to define the guidance philosophy, guidance control variables, targets, etc., at each guidance event.

Method: Since the number of guidance events considered for a given SIMSEP run has been specified by the NGUID variable which was read in SDAT1, the SDAT2 subroutine reads the $GUID namelist NGUID-times. Names, dimensions, default values, and definitions for the variables contained in $GUID are discussed in the User's Manual (Section 2.4, page 37). The input data from $GUID are stored in common blocks for subsequent usage during the execution of guidance maneuvers. The user specifies through input the type of guidance, duration of the guidance event, target variables and controls.

Remarks: Variables appearing in SDAT2 are initialized from namelist in external "user" units. As was done in SDAT1, these variables are converted to internal units before being transmitted to the rest of the program.

Input/Output:

Variable	Input/ Output	Namelist/ Common	Definition
TGUID	I	N	Guidance event epoch
TGE	0	C	
XGREF	I	N	Reference trajectory state vector at the guidance point.
RXGE	0	C	
MGREF	I	N	S/C mass at the guidance point.
RMGE	0	C	
S	I	N	Sensitivity or guidance matrix.
SMAT	0	C	
H	I	N	Array of on/off flags used to identify active thrust controls at a guidance event.
IJH	0	C	Matrix of active control variable indices.

Input/Output:

Variable	Input/ Output	Namelist/ Common	Definition
UWATE	I	N	Control variable weights.
CØNWT	0	C	
IGUID	I	N	Guidance law flag.
IGL	0	C	
NMAX	I	N	Maximum number of iterations in the nonlinear guidance algorithm.
ITMX	0	C	
NEP	I	N	Ephemeris planet number.
IEPH	0	C	
NTP	I	N	Target planet number.
NTPL	0	C	
NTAR	0	C	Number of target variables.
NTC	0	C	Number of control variables.
TTARG	I	N	Target epoch.
TTAR	0	C	
TARGET	I	N	Target variables evaluated on the reference trajectory.
XTARG	0	C	
XTREF	I	N	Reference trajectory state at the target epoch
RXTAR	0	C	
MTREF	I	N	S/C mass at the target epoch.
RMTAR	0	C	
TARTØL	I	N	Target variable tolerances.
TØL	0	C	
ITARGT	I	N	Target variable selection flags.
LSTAR	0	C	

Input/Output:

Variable	Input/ Output	Namelist/ Common	Definition
P	I	N	Augmented knowledge error covariance at a guidance event.
PS	I	N	
CXS	I	N	
BLANK	0	C	Eigenvectors and eigenvalues.
KDIMEN	I	N	Dimension of the augmented knowledge covariance.
KDIM	0	C	
KTER	I	N	Option flag for computing target errors.
KTERR	0	C	
CCØVG	I	N	Accumulated control error statistics at the guidance point.
GCCØV	0	C	
GMSCØV	I	N	Accumulated S/C mass statistics at the guidance point.
GMCØV	0	C	
CNTCØV	I	N	Accumulated active thrust control error statistics.
GNCØV	0	C	
DVMCOV	I	N	Accumulated delta-velocity vector statistics at the guidance event.
DVCOV	0	C	
DVMAG	I	N	Accumulated delta-velocity magnitude statistics at the guidance event.
DVMAGS	0	C	
CCØVT	I	N	Accumulated control error statistics at the target point.
TCCØV	0	C	
TMSCØV	I	N	Accumulated S/C mass statistics at the target point.
TMCOV	0	C	
TARCØV	I	N	Accumulated target error statistics.
TERCØV	0	C	

Input/Output:

Variable	Input/ Output	Namelist/ Common	Definition
MSAMP	I	N	Number of previous Monte Carlo samples on the accumulated statistics.
NSAMP	O	C	
MTPH	O	C	Thrust phase number at a guidance event.
ICYCLE	O	C	Recycle flag.
UNTAR	O	C	Vector of target variable conversion factors.

Local Variables:

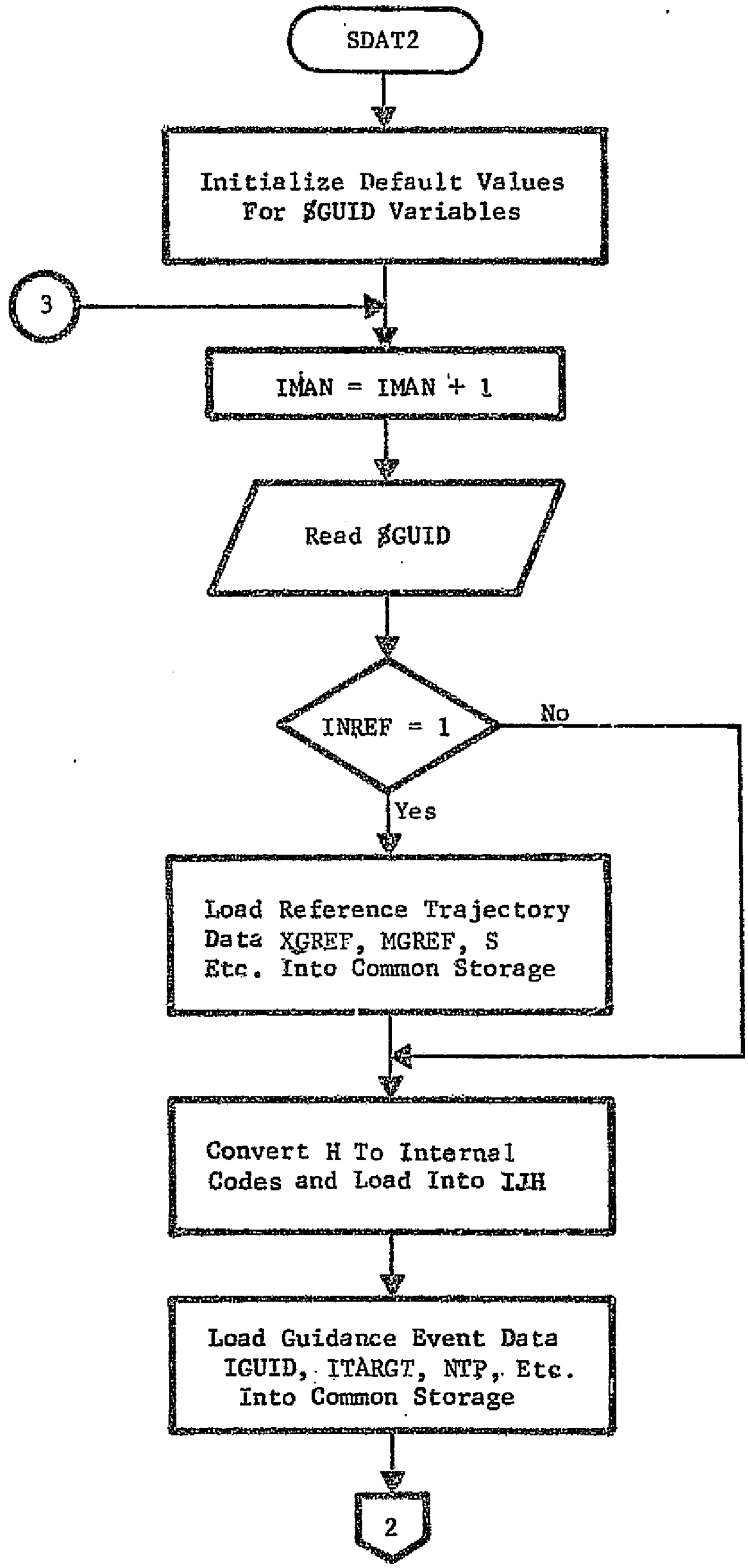
Variable	Definition
PP	Temporary storage for the augmented knowledge covariance matrix.
IMAN	Guidance event counter.
INDEX1	Index marking the position in blank common after which eigenvectors corresponding to a particular augmented knowledge covariance are stored.
INDEX2	Index like INDEX1 except it marks where eigenvalues are stored.

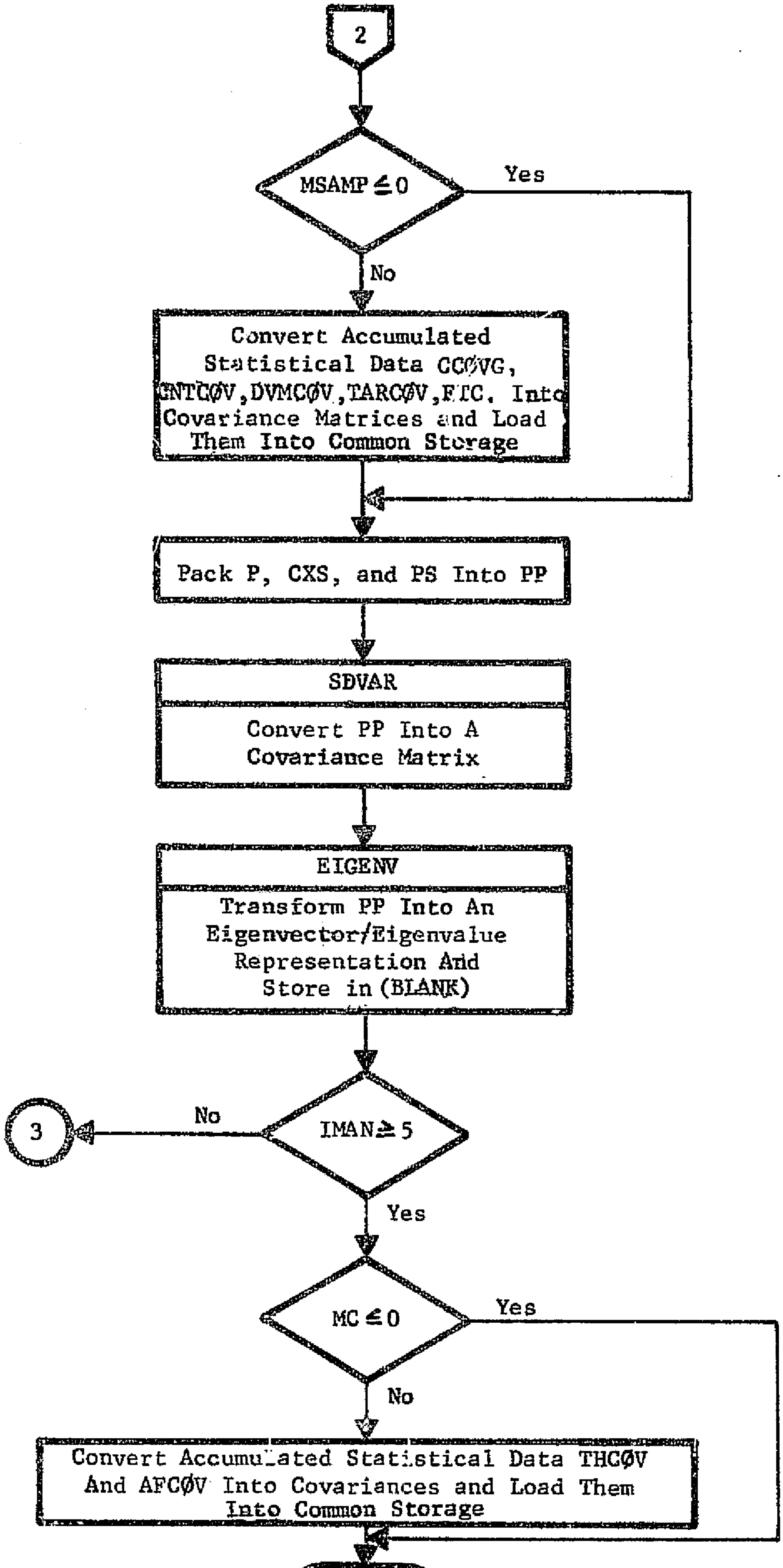
Subroutines Called: CØPY, EIGENV, ICØPY, MATØUT, MPAK, MUNPAK, SDVAR, SYMLØ, SYMUP, ZERØM.

Calling Subroutines: DATAS

Common Blocks: CYCLE, EDIT, EPHEM, IASTM, SIM1, ISIM1, SIM2, ISIM2, SIMLAB, TIME, TRAJ1, TRAJ2, (BLANK).

Logic Flow:


SDAT2
Initialize Default Values For $GUID Variables
3
IMAN = IMAN + 1
Read $GUID
INREF = 1
No
Yes
Load Reference Trajectory Data XGREF, MGREF, S Etc. Into Common Storage
Convert H To Internal Codes and Load Into IJH
Load Guidance Event Data IGUID, ITARGT, NTP, Etc. Into Common Storage
2


2
MSAMP ≦ 0
Yes
No
Convert Accumulated Statistical Data CCØVG, CNTCØV, DVMCØV, TARCØV, ETC. Into Covariance Matrices and Load Them Into Common Storage
Pack P, CXS, and PS Into PP
SDVAR
Convert PP Into A Covariance Matrix
EIGENV
Transform PP Into An Eigenvector/Eigenvalue Representation And Store in (BLANK)
IMAN ≧ 5
No
3
Yes
MC ≦ 0
Yes
No
Convert Accumulated Statistical Data THCØV And AFCØV Into Covariances and Load Them Into Common Storage

3.4.10A Subroutine: SET (ISTØRE)

Purpose: To set and store physical parameters (ephemeris, gravitational, etc.) and SEPS parameters (thrust controls, mass, exhaust velocity, etc.) needed by the trajectory integration routine for generating the actual, estimated, and reference trajectories.

Method: SET simply performs multiple copy operations in transferring the working values used by the trajectory integrator into designated storage arrays, S1, S2 and S3. By calling SET with ISTØRE equal to +1, +2 or +3, the corresponding S1, S2 or S3 array is equated to whatever is in the regular working arrays. If ISTØRE equals +4, all three S-arrays are set. When SET is called with ISTØRE equal to -1, -2, or -3, then the working arrays are re-set to whatever is stored in S1, S2 or S3, respectively.

Remarks: This routine is essential to SIMSEP in that it allows the program to use the same trajectory integrator to evaluate each of the different types of trajectories needed for a mission simulation.

Input/Output:

Variables	Input/ Output	Argument/ Common	Definition
ISTØRE	I	A	Flag controlling the SET logic flow.
ENGINE(1) (=PØWERO)	I/O	C	Electric power at 1 A.U.
ENGINE(10) (=EXHVEL)	I/O	C	Thrust exhaust velocity.
ENGINE(11) (=THREFF)	I/O	C	Thruster efficiency.
ENGINE(15) (=CRA)	I/O	C	Radiation pressure coefficient.
SCMASS	I/O	C	SEPS mass.
SMASS	I/O	C	Solar gravitational constant.
PMASS	I/O	C	Planetary gravitational constants.
NTPHAS	I/O	C	Current thrust control phase number.
XEPH	I/O	C	Ephemeris planet state vector.
THRUST	I/O	C	Thrust control array.
CSAX	I/O	C	Semi-major axis ephemeris constants.
EMN	I/O	C	Lunar ephemeris constants.
CECC	I/O	C	Eccentricity ephemeris constants.
CINC	I/O	C	Inclination ephemeris constants.
CØMEG	I/O	C	Longitude of the ascending node ephemeris constants.

Variable	Input/ Output	Argument/ Common	Definition
CØMEGT	I/O	C	Argument of the apsis ephemeris constants.
CMEAN	I/O	C	Mean anomaly ephemeris constants.
SSM1	I/O	C	Stored solar gravitational constant.
SSCM1	I/O	C	Stored SEPS mass.
SEXV1	I/O	C	Stored thrust exhaust velocity.
STEFF1	I/O	C	Stored thruster efficiency.
SCRA1	I/O	C	Stored radiation pressure.
SPO1	I/O	C	Stored electric power to 1. A.U.
SPM1	I/O	C	Stored planetary gravitational constants.
SXEPH1	I/O	C	Stored ephemeris planet state vectors.
SSAX1	I/O	C	Stored semi-major axes.
SEMN1	I/O	C	Stored lunar ephemeris constants.
SECC1	I/O	C	Stored eccentricities.
SINC1	I/O	C	Stored inclinations.
SØMEG1	I/O	C	Stored longitudes of the ascending node.
SØMGT1	I/O	C	Stored arguments of the apsis.
SMEAN1	I/O	C	Stored mean anomalies.
STHRT1	I/O	C	Stored thrust controls.

(Comment: In addition to these storage arrays and variables, there are also corresponding S-2 and S-3 arrays.)

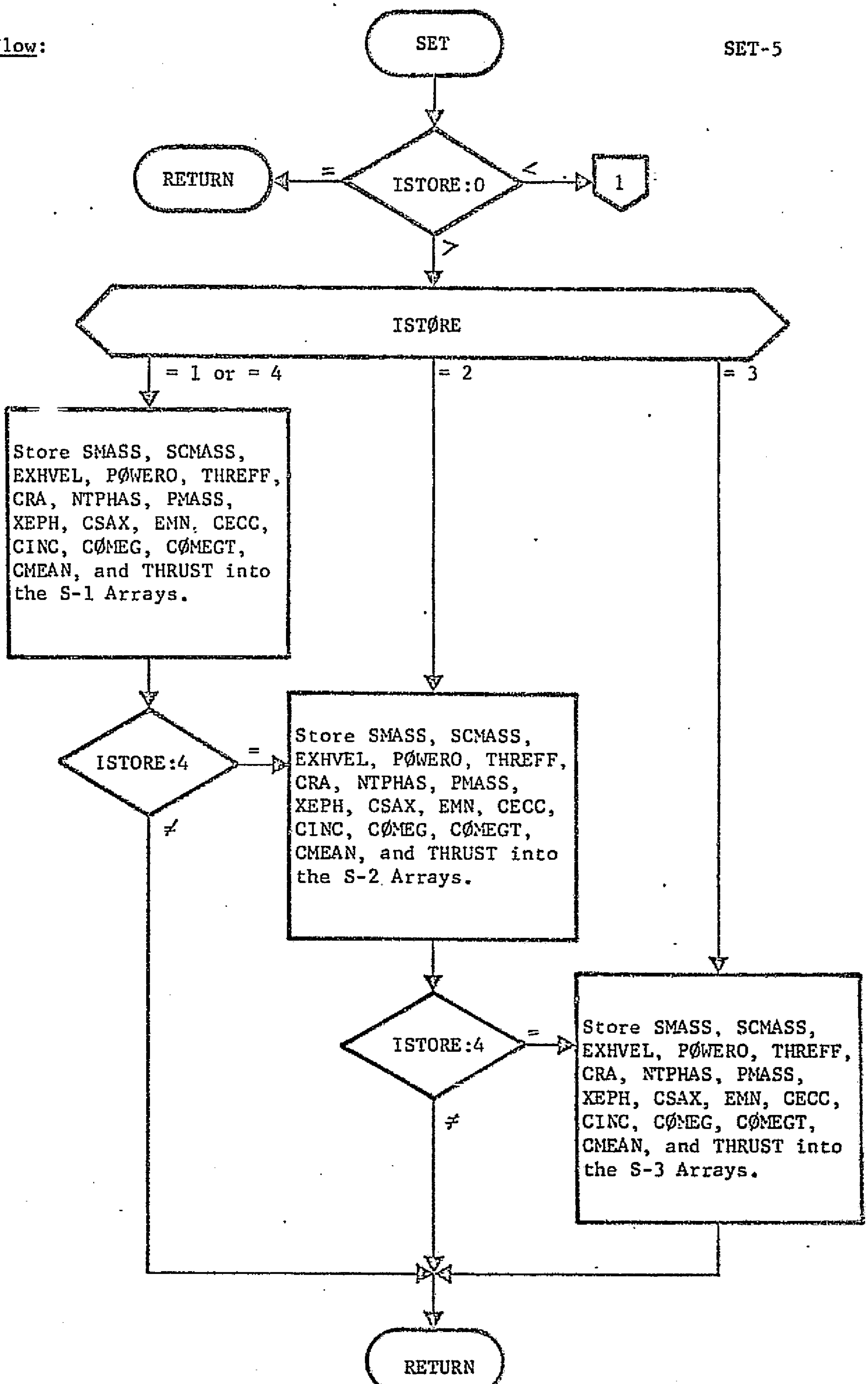
Local Variables: None

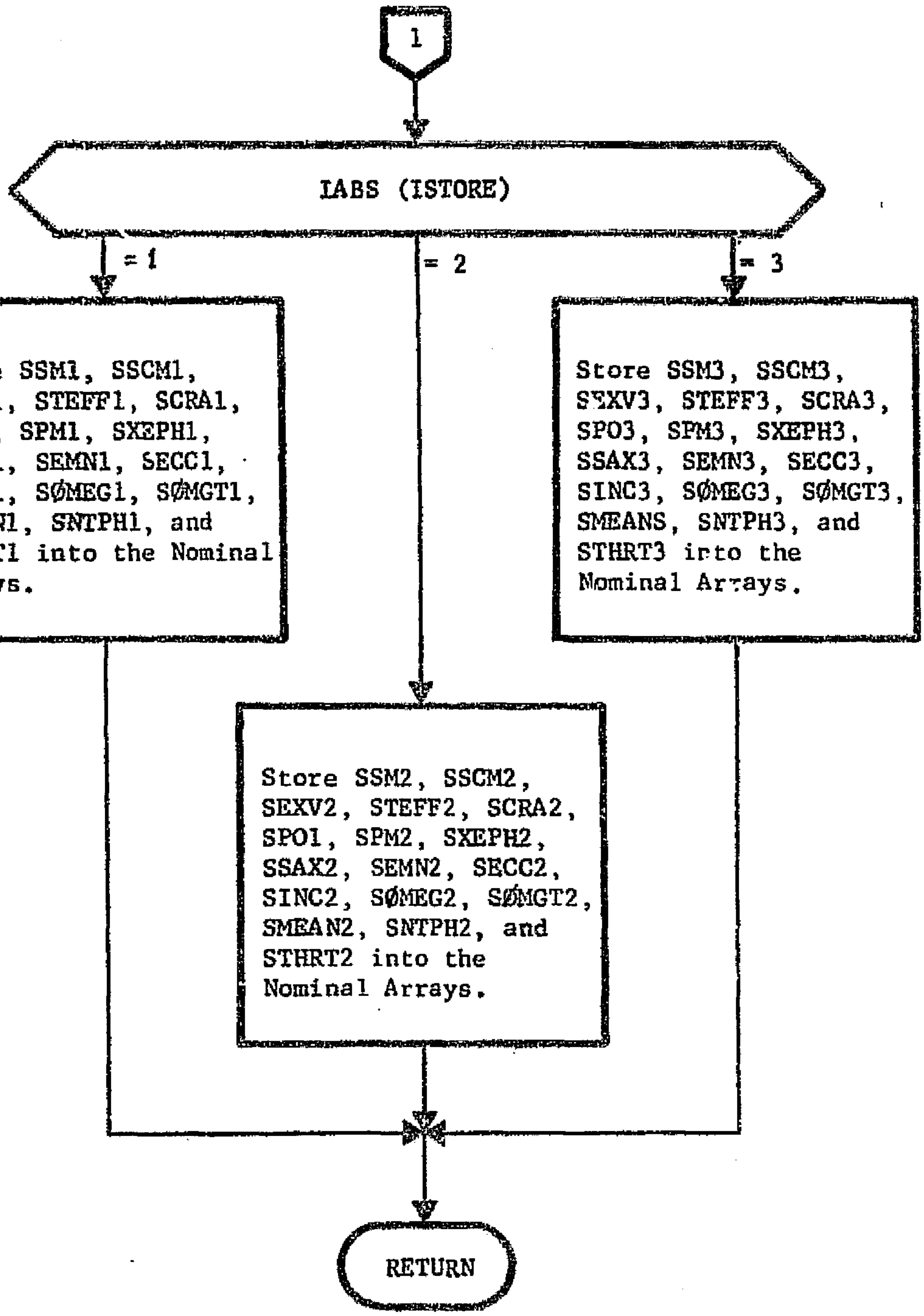
Subroutines Called: CØPY

Calling Subroutines: SIMSEP, NLGUID

Common Blocks: EPHEM, SIM1, ISIM1, STØREC, TRAJ1, TRAJ2

Logic Flow:


SET
ISTORE:0
RETURN
=
<
>
1
ISTØRE
= 1 or = 4
= 2
= 3
Store SMASS, SCMASS, EXHVEL, PØWERO, THREFF, CRA, NTPHAS, PMASS, XEPH, CSAX, EMN, CECC, CINC, CØMEG, CØMEGT, CMEAN, and THRUST into the S-1 Arrays.
ISTORE:4
=
≠
Store SMASS, SCMASS, EXHVEL, PØWERO, THREFF, CRA, NTPHAS, PMASS, XEPH, CSAX, EMN, CECC, CINC, CØMEG, CØMEGT, CMEAN, and THRUST into the S-2 Arrays.
ISTORE:4
=
≠
Store SMASS, SCMASS, EXHVEL, PØWERO, THREFF, CRA, NTPHAS, PMASS, XEPH, CSAX, EMN, CECC, CINC, CØMEG, CØMEGT, CMEAN, and THRUST into the S-3 Arrays.
RETURN


1
IABS (ISTORE)
= 1
= 2
= 3
Store SSM1, SSCM1, SEXV1, STEFF1, SCRA1, SPO1, SPM1, SXEPH1, SSAX1, SEMN1, SECC1, SINC1, SØMEG1, SØMGT1, SMEAN1, SNTPH1, and STMRT1 into the Nominal Arrays.
Store SSM3, SSCM3, SEXV3, STEFF3, SCRA3, SPO3, SPM3, SXEPH3, SSAX3, SEMN3, SECC3, SINC3, SØMEG3, SØMGT3, SMEANS, SNTPH3, and STHRT3 into the Nominal Arrays.
Store SSM2, SSCM2, SEXV2, STEFF2, SCRA2, SPO1, SPM2, SXEPH2, SSAX2, SEMN2, SECC2, SINC2, SØMEG2, SØMGT2, SMEAN2, SNTPH2, and STHRT2 into the Nominal Arrays.
RETURN

3.4.10B Subroutine: SPRNT1 (XA, XE, XREFO, IC, IMAN)

Entry Points: SPRNT2, SPRNT3, SPRNT4

Purpose: To print actual, estimated, and reference trajectory data computed during Monte Carlo mission simulations.

Method: SPRNT1, or one of its various entry points, is called from SIMSEP whenever printout of trajectory information is desired. A call to SPRNT1 results in the "Output Data for the Actual Trajectory Initialization". (See the sample case in the User's Manual, Pages 119 through 132.) SPRNT2 generates the "Output Data for Guidance Event" which includes printout for actual, estimated, and reference trajectory data. SPRNT3 generates the "Output Data at the Designated Target Time" when KTER = 1 and the corrected trajectory is propagated after a guidance event. At the end of each Monte Carlo mission simulation, SPRNT4 is called to display the "Monte Carlo Mission Summary".

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
XA	I	A	Current actual S/C state.
XE	I	A	Current estimated S/C state.

Variable	Input/ Output	Argument/ Common	Definition
XREFO	I	A	Current reference S/C state.
IC	I	A	Current Monte Carlo cycle number.
IMAN	I	A	Current guidance event number.
TCERR	I	C	Thrust bias errors.
SCERR	I	C	S/C and SEP errors.
GMERR	I	C	Gravitational constant errors.
GTAU1	I	C	Negative reciprocal of the correlation times for the thrust process noise.
GTAU2	I	C	
TNØISE	I	C	Vector of random thrust control perturbations.
TGE	I	C	Guidance event epoch.
TTAR	I	C	Target epoch.
XTARG	I	C	Reference target variables.
UNTAR	I	C	Vector of target variable conversion factors.
SSCM1	I	C	Actual, estimated, and reference S/C mass.
SSCM2	I	C	
SSCM3	I	C	
SEXV1	I	C	Actual, estimated, and reference exhaust velocity.
SEXV2	I	C	
SEXV3	I	C	
SPO1	I	C	Actual, estimated, and reference electric power.
SPO2	I	C	
SPO3	I	C	
STEFF1	I	C	Actual, estimated, and reference thruster efficiency.
STEFF2	I	C	
STEFF3	I	C	

Variable	Input/ Output	Argument/ Common	Definition
SCRA1	I	C	Actual, estimated, and reference radiation pressure co-efficient.
SCRA2	I	C	
SCRA3	I	C	
STHRT1	I	C	Actual, estimated, and reference thrust controls.
STHRT2	I	C	
STHRT3	I	C	
SSM1	I	C	Actual, estimated and reference solar gravitational constant.
SSM2	I	C	
SSM3	I	C	
SXEPM1	I	C	Actual, estimated and reference Cartesian state for the ephemeris body at TEPH.
SXEPM2	I	C	
SXEPM3	I	C	
SPM1	I	C	Actual, estimated, and reference gravitational constant for the ephemeris body.
SPM2	I	C	
SPM3	I	C	

Local Variables:

Variable	Definition
DXE	Vector deviation of the estimated state from the reference and/or the actual.
DXA	Vector deviation of the actual state from the reference.
ELA	Keplerian elements corresponding to the actual, estimated, and reference Cartesian states of ephemeris body.
ELE	
ELR	
EMASS	Actual S/C mass evaluated at TEND.

Subroutines Called: CØNIC, SUB

Calling Subroutines: SIMSEP

Common Blocks: CØNST, DYNØS, EPHEM, SIM1, ISIM1, SIMLAB, STØREC, TIME, TRAJ1, TRAJ2, (BLANK)

Logic Flow: None.

3.4.11 Subroutine: STAT (XA, XR, N, N1, ACØV, M, PCØV)

Purpose: To compute a covariance matrix and mean, recursively, from a sequence of error vectors.

Method: For the M^{th} Monte Carlo cycle, an error vector, X_M, is computed as the difference between an actual and a reference vector. This error vector updates the previous mean based on (M-1) samples according to the equation

$$\overline{X}_M = (X_M + (M-1)\,\overline{X}_{M-1})/M$$

for M = 1, 2, 3, ... The covariance matrix is also updated by the relation,

$$C_M = \left[\frac{M-2}{M-1}\right] C_{M-1} + \left[\overline{X}_{M-1}\,\overline{X}^T_{M-1}\right]$$

$$+ \frac{1}{M-1} X_M X_M^T - \frac{M}{M-1} \overline{X}_M \overline{X}_M^T$$

for M = 2, 3, 4, ..., where C_{M-1} is the previous covariance matrix and C_M the new covariance.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
XA	I	A	Actual sampled vector.
XR	I	A	Reference vector.
N	I	A	Dimension of XA and XR.
N1	I	A	N1 = N + 1.

Variable	Input/ Output	Argument/ Common	Definition
ACØV	I	A	A-prior covariance matrix and mean, based on M-1 samples. This is a (NxN1) array with the variances and covariances being stored in the first N-columns and the means being stored in the N1-columns.
M	I	A	Number of Monte Carlo samples used to formulate the updated covariance matrix.
PCØV	I	A	Updated output covariance matrix and vector of means. The storage is in the same format as ACØV. ACØV and PCØV may, in fact, share the same core locations.

Local Variables:

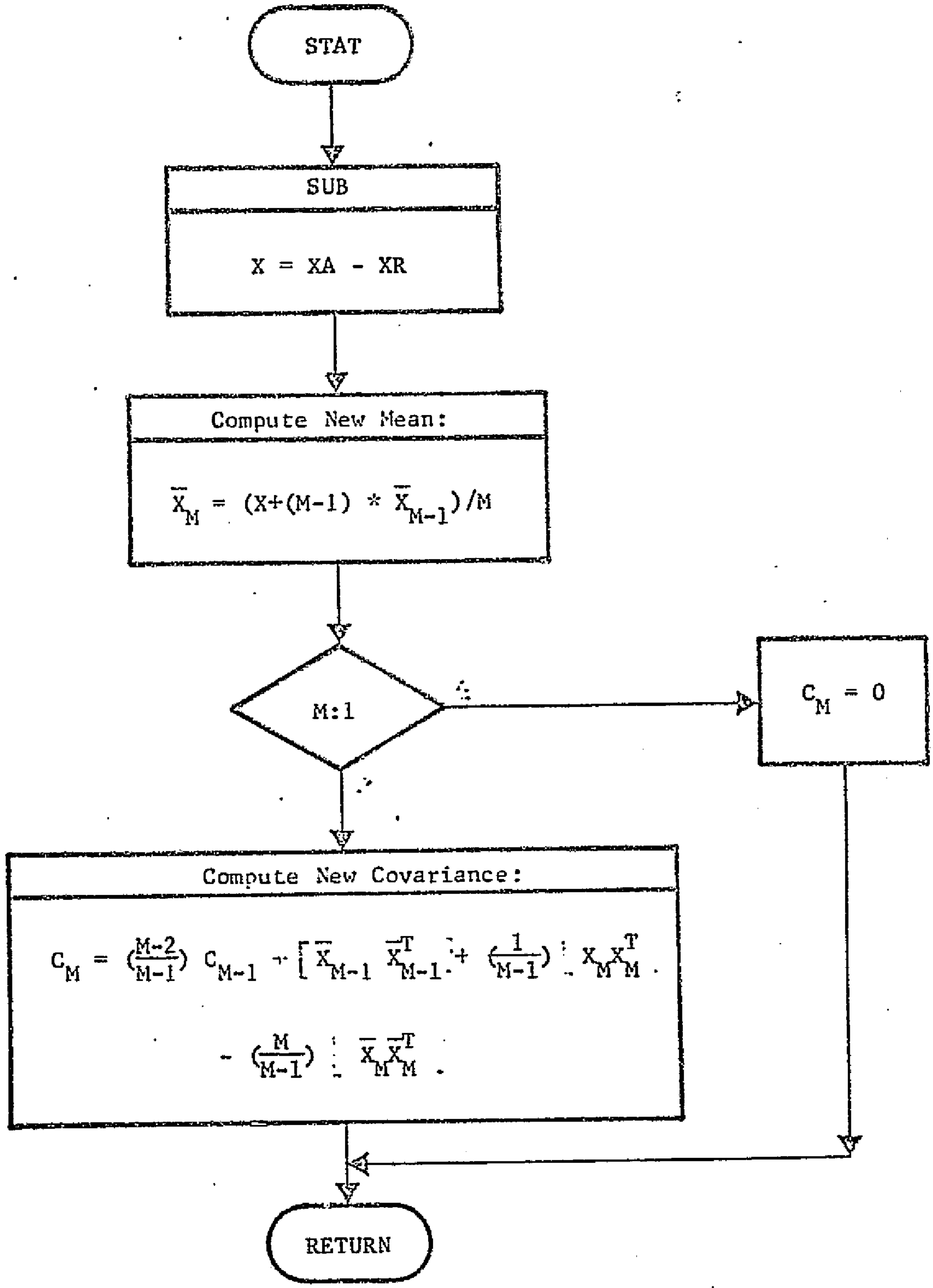
Variable	Definition
X	Error vector, X = XA - XR.
XX	Temporary storage for the new means.
XXT	Temporary storage for the outer product of two vectors.

Subroutines Called: SUB

Calling Subroutines: SIMSEP

Common Blocks: WØRK

Logic Flow:


STAT
SUB
X = XA - XR
Compute New Mean:
$\bar{X}_M = (X+(M-1) * \bar{X}_{M-1})/M$
M:1
$C_M = 0$
Compute New Covariance:
$C_M = (\frac{M-2}{M-1}) C_{M-1} + [\bar{X}_{M-1} \bar{X}^T_{M-1} + (\frac{1}{M-1}) X_M X^T_M$
$- (\frac{M}{M-1}) \bar{X}_M \bar{X}^T_M$
RETURN

3.4.12 Subroutine: THCØMP (XE, XXE, TSAVE, IMAN, THETA)

Purpose: To compute the matrix of partial deviatives of state coordinates at the target with respect to thrust controls; namely,

$$\Theta = \left[\frac{\partial (X, Y, Z, \dot{X}, \dot{Y}, \dot{Z})}{\partial (u_1, u_2, \ldots, u_n)} \right] \quad (6 \times n) \quad ,$$

where n is the number of control variables.

Method: Small perturbations are forced to each control variable of concern and an integrated trajectory is propagated to the target time. The final state vector of each variant trajectory is differenced with the standard, or nominal, state to form numerical partials.

Remarks: This subroutine is used by both the linear and nonlinear guidance subroutines in SIMSEP. Conversion of THCØMP to use augmented state transition matrices integrated during the trajectory propagation would principally affect this routine and would considerably reduce the computational expense of numerically differencing.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
XE	I	A	Initial state vector which is to be perturbed in computing the partials.

Variable	Input/ Output	Argument/ Common	Definition
XXE	I	A	Final state vector on the standard as nominal trajectory.
TSAVE	I	A	Trajectory propagation interval between the guidance event and the target time.
IMAN	I	A	Number of the current guidance event.
THETA	O	A	Θ - matrix of partial derivatives.
THRUST	I/O	C	Thrust control array.
IHMAT	I	C	Array of indices that specify the thrust controls which are to be perturbed.
HMAT	I	C	Array of thrust control perturbations.
UREL	I	C	Target body relative position vector at the target time.
VREL	I	C	Target body relative velocity vector at the target time.
NTCØN	I	C	Number of perturbing controls.

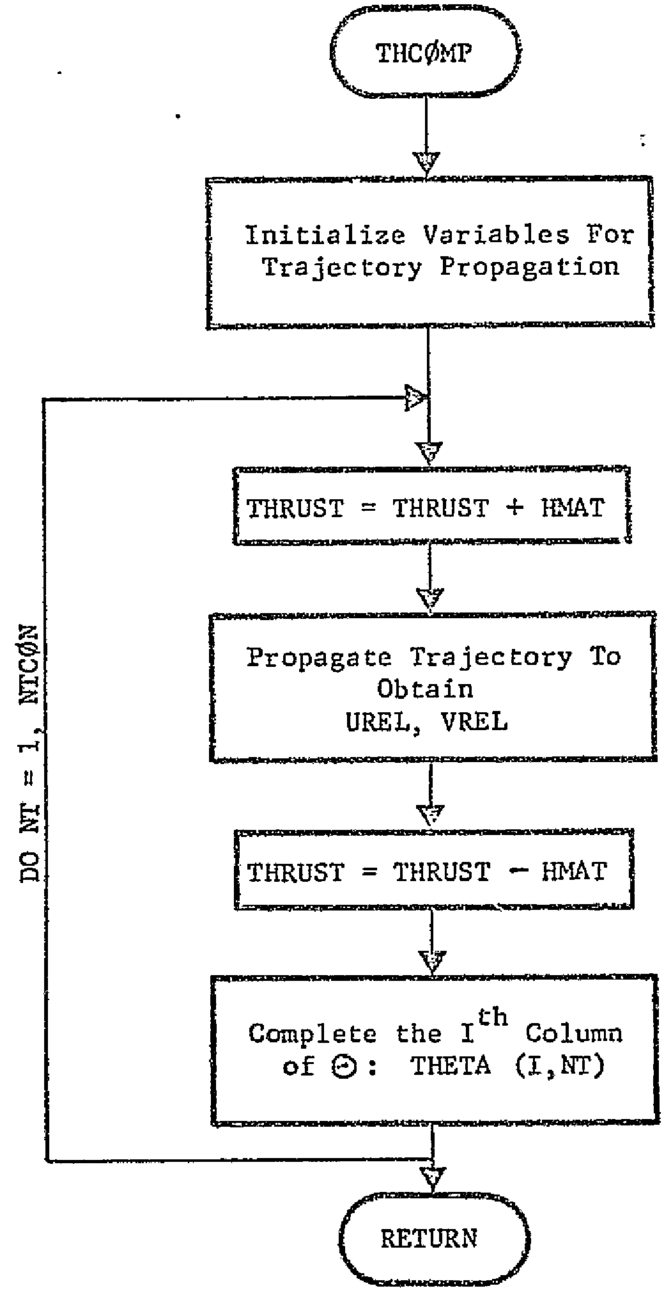
Local Variables: None

Subroutines Called: TRAJ, CØPY

Calling Subroutines: LGUID, NLGUID

Common Blocks: CØNST, TRAJ1, TRAJ2, TIME, SIM1

Logic Flow:

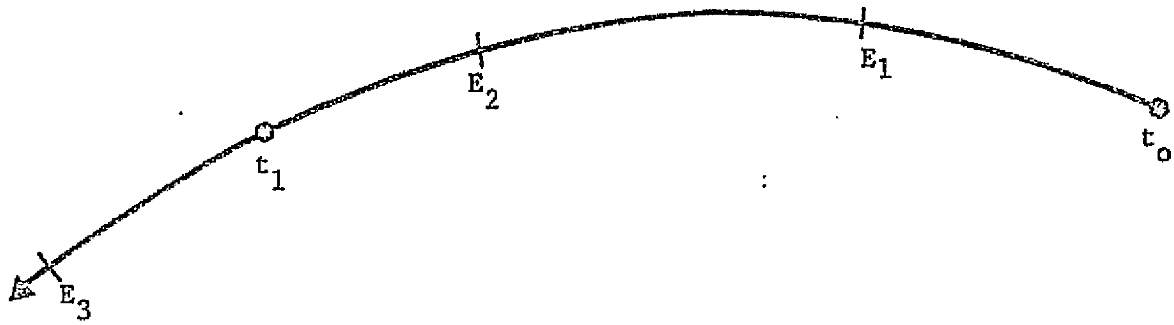

THCØMP
Initialize Variables For Trajectory Propagation
DO NT = 1, NTCØN
THRUST = THRUST + HMAT
Propagate Trajectory To Obtain UREL, VREL
THRUST = THRUST − HMAT
Complete the I th Column of Θ: THETA (I,NT)
RETURN

3.5 Subroutine: TRAJ

Purpose: To control the overall trajectory initialization and propagation.

Remarks: Since TRAJ is used by the three modes, it must be capable of reproducing the same trajectory for each mode, independent of the augmented state form, event times or print times. Special problems arise when the equations to be propagated include the transition matrix or covariance between events. For example, at the beginning of an event either the transition matrix must be reset to an identity or an updated covariance must be given to TRAJ. To solve these problems, logic was incorporated into TRAJ to make use of event logic in the subroutine PATH with an entry point FLIGHT.

Beginning at the trajectory epoch t_o, the transition matrix or covariance is initialized and is propagated to the first event (E_1). MAPSEP logic returns to the calling routine which performs its operations. Upon reentering TRAJ, the transition matrix or covariance is again reinitialized and



propagated from E_1 to E_2. In order to propagate the transition matrix or covariance from E_2 to E_3 and preserve the trajectory grid, the special logic in TRAJ calls FLIGHT to propagate the appropriate matrix from E_2 to t_1. Then the spacecraft state is propagated from t_o to t_1. Now having the state and transition matrix or covariance at t_1, the appropriate matrix is propagated to E_3. This process is continued until all events have been satisfied.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
ICALL	I	C	= 1, initialize the trajectory and propagate to an event or stopping condition. = 2, initialize the trajectory only. = 3, propagate from a previously defined point in the trajectory.
INTEG	I	C	= 1, propagate the state and transition matrix. = 2, propagate the state. = 3, propagate the state and state covariance matrix.

Variable	Input/ Output	Argument/ Common	Definition
DSC	I	C	The blank common array. The following flags will be used to locate specific information.
LØCET	I	C	Previous event.
LØCX	I	C	Trajectory time.
LØCH	I	C	Integration stepsize.
LØCTC	I	C	State transition matrix or Covariance.
LØCFØ	I	C	Deviations (from conic) of state (reference).
LØCDY	I	C	Deviations (from conic) of state derivatives (reference).
LØCYT	I	C	Deviations of state (event).
LØCDT	I	C	Deviations of state derivatives (event).
MEQS	I	C	Dimensions of the covariance or transition matrix.
TEVNT	I	C	Next event time.
IAUGDC	I	C	Flags used to augment the covariance or transition matrix.

Local Variables:

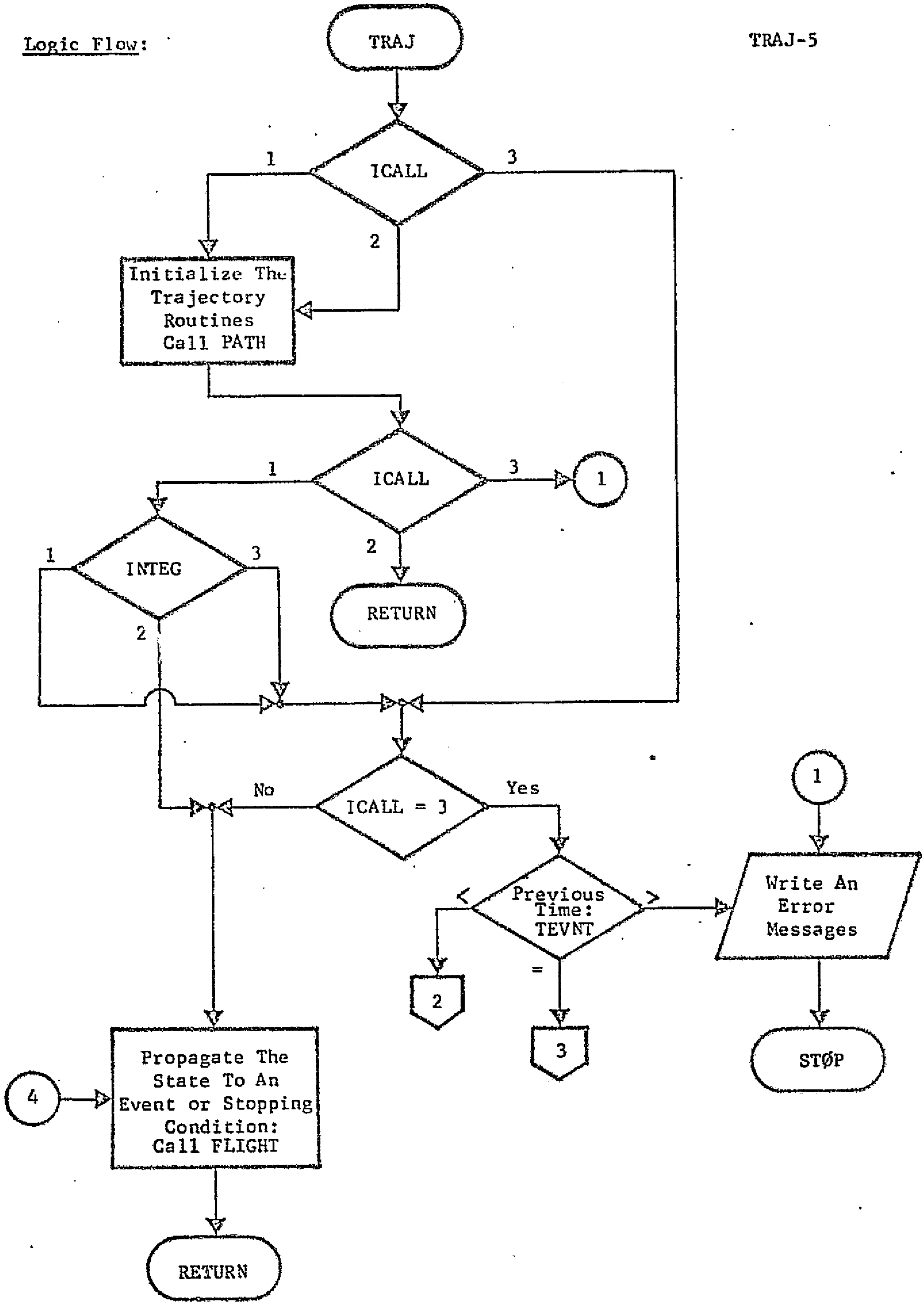
Variable	Definition
TEVNTS	Stored value of TEVNT.
IAUGDS	Stored value of IAUGDC.

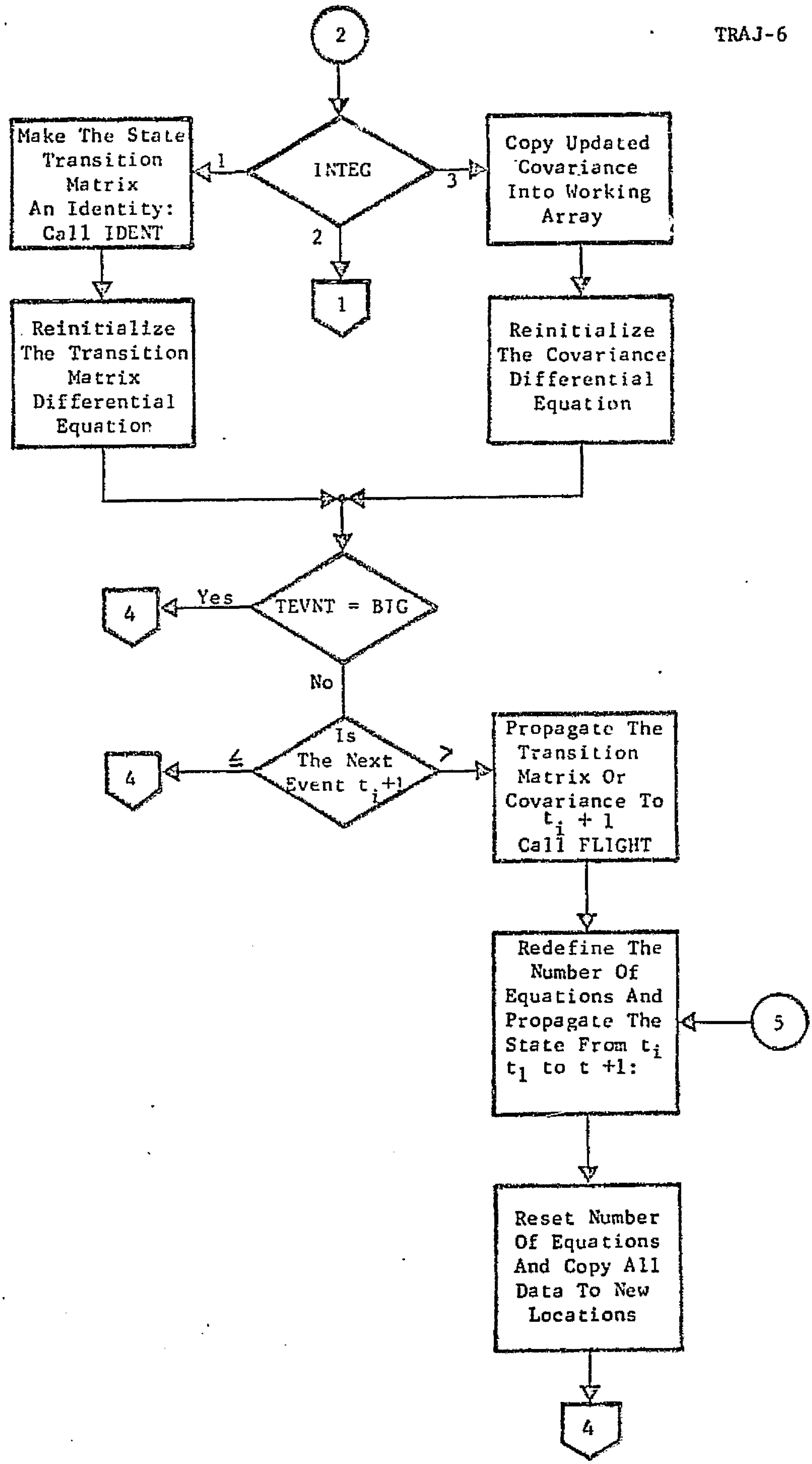
Subroutines Called: PATH, FLIGHT, IDENT, CØPY, LØADFM

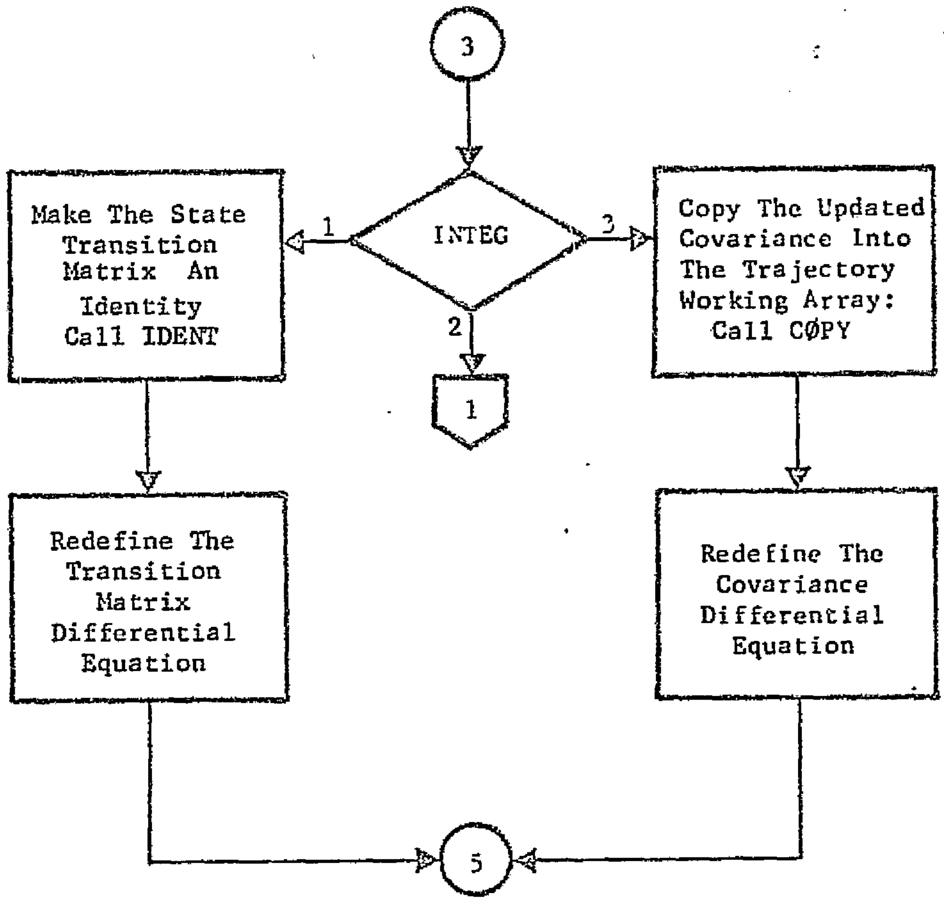
Calling Subroutines: TØPSEP, GØDSEP, SIMSEP

Common Blocks: TRAJ2, WØRK, (BLANK), CØNST, EDIT, EPHEM, TIME, TRAJ1

<u>Logic Flow:</u>




2
INTEG
1
Make The State Transition Matrix An Identity: Call IDENT
3
Copy Updated Covariance Into Working Array
2
1
Reinitialize The Transition Matrix Differential Equation
Reinitialize The Covariance Differential Equation
TEVNT = BIG
Yes
4
No
Is The Next Event t_i+1
≤
4
>
Propagate The Transition Matrix Or Covariance To t_i + 1 Call FLIGHT
Redefine The Number Of Equations And Propagate The State From t_i t_1 to t +1:
5
Reset Number Of Equations And Copy All Data To New Locations
4


3
INTEG
1
2
3
Make The State Transition Matrix An Identity Call IDENT
1
Copy The Updated Covariance Into The Trajectory Working Array: Call CØPY
Redefine The Transition Matrix Differential Equation
Redefine The Covariance Differential Equation
5

3.5.1 Subroutine: DNØISE (T)

Entry Point: NØISE

Purpose: To compute thrust acceleration perturbations due to time-varying process noise.

Method: A vector of thrust control perturbations, $\delta \underline{u}$, is computed during the trajectory integration at the beginning, middle, and end of each integration step. The time correlated thrust noise is assumed to be a Gauss-Markov sequence according to the equation

$$\delta \underline{u}_{i+1} = A\, \delta \underline{u}_i + \underline{\omega}_{i+1} ,$$

where

$$A = \begin{bmatrix} e^{-\Delta t/T_1} & & & 0 \\ & e^{-\Delta t/T_2} & & \\ & & \ddots & \\ 0 & & & e^{-\Delta t/T_N} \end{bmatrix}$$

and $\Delta t = t_{i+1} - t_i$. The factors $T_1, T_2, \ldots, T_N$ are the correlation times associated with each stochastic process, δu_j. The vector $\delta \underline{u}_i$ is assumed to remain constant over the interval Δt with its effect on $\delta \underline{u}_{i+1}$ being diminished by the exponential decay terms in A. $\underline{\omega}_{i+1}$ is a vector of independent random variables which have Gaussian distributions. The standard deviation, σ_{ω_j}, is given by

$$\sigma_{\omega_j} = (1-e^{-2\Delta t/T_j})^{\frac{1}{2}} \sigma_{u_j} ,$$

in order to satisfy the requirement that the process be stationary.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
T	I	A	Current trajectory time.
GTAU1	I	C	Negative reciprocal of the correlation times for the first process.
GTAU2	I	C	Negative reciprocal of the correlation times for the second process.
TVERR	I	C	One-sigma values for the time-varying thrust control errors.
IRAN	I	C	Random number seed.
TNØISE	I/O	C	Vector of thrust control perturbations.

Local Variables:

Variable	Definition
T1	Trajectory time at the previous point of thrust noise evaluation.

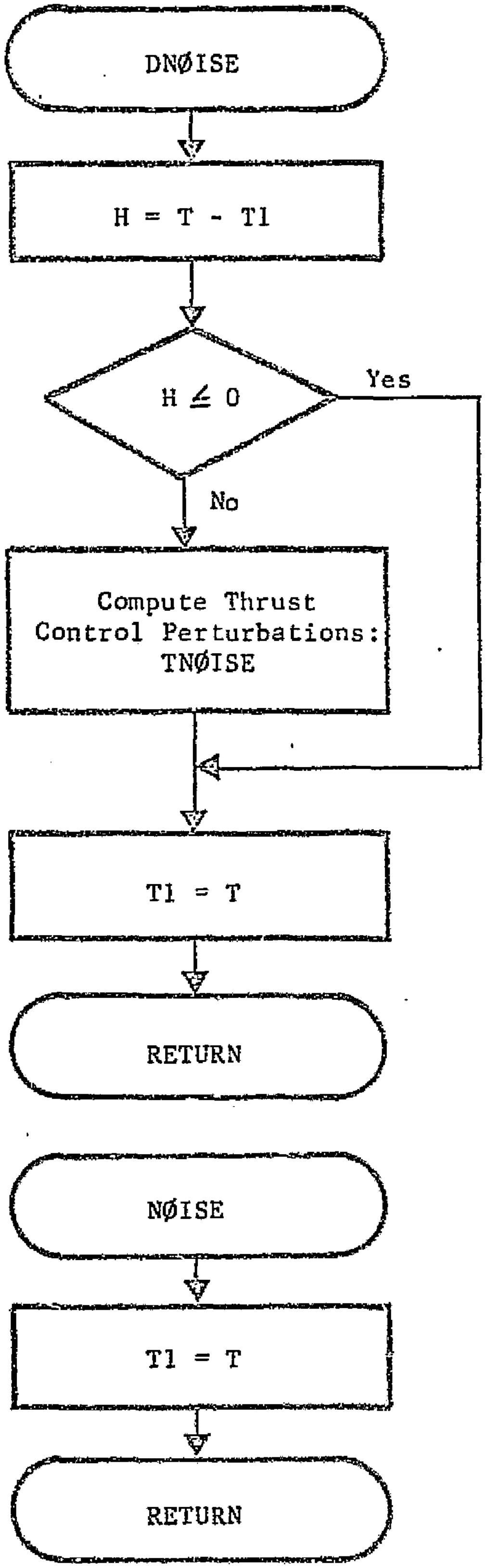
Variable	Definition
H	Time increment since the previous thrust noise evaluation.

Subroutines Called: RNUM

Calling Subroutines: EP, SIMSEP

Common Blocks: TRAJ1, DYNØS, TRAJ2

Logic Flow:


DNØISE
H = T - T1
H ≤ 0
Yes
No
Compute Thrust Control Perturbations: TNØISE
T1 = T
RETURN
NØISE
T1 = T
RETURN

3.5.2 Subroutine: DPHI (T, DS, DSTM, M, N, LØC)

Purpose: To compute the time derivative of the State Transition Matrix (ϕ)

Method: $\dot{\phi} = F\phi$

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
IAUGDC	I	C	Flag indicating the augmentation of the STM and covariance Matrix.
T	I	A	Trajectory time
DS	I	A	Independent variables
DSTM	O	A	Differential equations
M	I	A	Number of rows in DS and DSTM
N	I	A	Number of columns in DS and DSTM
LO¢	I	A	Routing flag
INTEG	I	C	Set = 1 Propagate the State and Transition Matrix Set = 2 Propagate the State Set = 3 Propagate the State and State Covariance
IRECT	I	C	Index used to check whether the current call to DPHI is for rectification purposes only (i.e. IRECT = 1)

Local Variables:

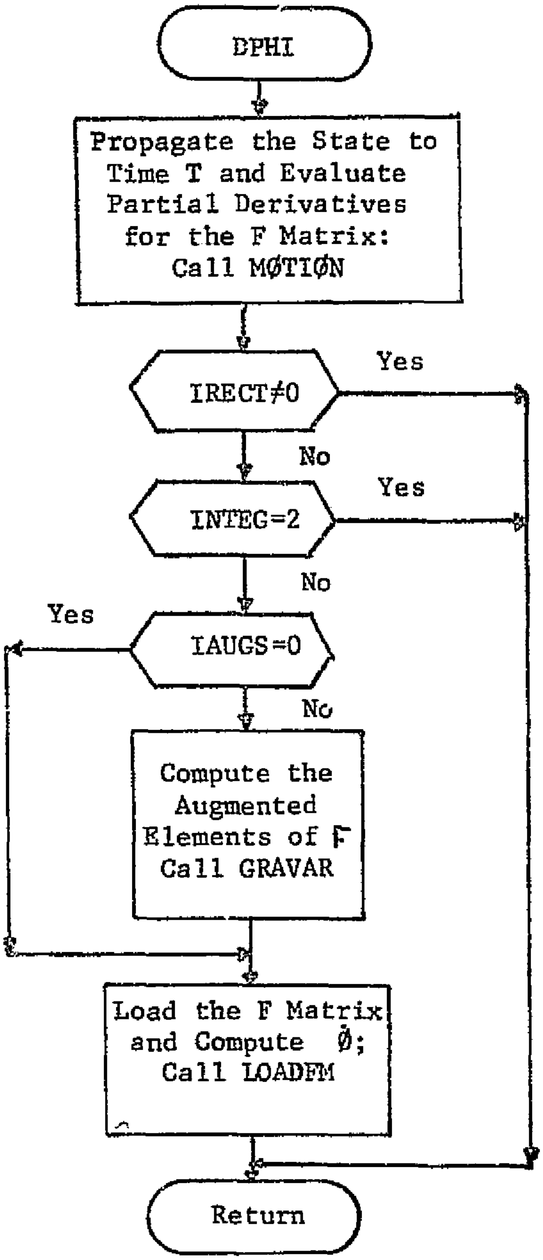
IAUGS Index used to check whether the F matrix needs to be augmented.

Calling Subroutines: NUMIN

Subroutines Called: MØTIØN, LØADFM, GRAVAR

Common Blocks: TRAJ2

Logic Flow:


DPHI
Propagate the State to Time T and Evaluate Partial Derivatives for the F Matrix: Call MØTIØN
IRECT≠0
Yes
No
INTEG=2
Yes
No
IAUGS=0
Yes
No
Compute the Augmented Elements of F Call GRAVAR
Load the F Matrix and Compute Ø̇; Call LOADFM
Return

3.5.3 Subroutine: EP (T, CMASS)

Purpose: To compute the effective low thrust acceleration vector and matrix of partial deviatives for transition matrix or covariance propagation in a control phase.

Method: The magnitude of the low thrust acceleration is

$$a = \frac{0.002\ \eta\ P\ T_L}{m\ c}$$

where

P = power available to the thrusters

η = thruster efficiency

m = spacecraft mass

c = exhaust velocity

T_L = throttling level.

The acceleration vector can be expressed in one of two spacecraft centered coordinate systems. One system is the cone and clock system and the other is the In plane and Out of plane or orbit plane system, (Section 4.1, Reference 1). Letting

C_N = Cone Angle

C_L = Clock Angle

δ = In plane angle

Υ = Out of plane angle.

The acceleration vectors for each system are

$$a'_x = a \cos (C_L + \dot{C}_L t) \sin (C_N + \dot{C}_N t)$$

$$a'_y = a \sin (C_L + \dot{C}_L t) \sin (C_N + \dot{C}_N t)$$

$$a'_z = a \cos (C_N + \dot{C}_N t)$$

and

$$a'_x = a \cos (\gamma + \dot{\gamma} t) \cos (\delta + \dot{\delta} t)$$

$$a'_y = a \cos (\gamma + \dot{\gamma} t) \sin (\delta + \dot{\delta} t)$$

$$a'_z = a \sin (\gamma + \dot{\gamma} t)$$

where t is the trajectory time from the beginning of the control phase. The acceleration is then transformed from the spacecraft system to the inertial system by the matrix A (See Section 4.1 of the Analytic Manual),

$$\underline{a} = A \, \underline{a}'$$

EP also computes the matrix of partial derivatives

$$g = \frac{\partial \underline{a}}{\partial \underline{u}}$$

where $\underline{u}$ is the vector of thrust controls in the current segment, NTPHASE. The controls are throttling level and the two pointing angles (cone/clock or in/out of plane) which correspond to THRUST (3, NTPHAS), THRUST (4, NTPHAS) and THRUST (5, NTPHAS), respectively.

For the cone/clock system,

$$g' = \begin{bmatrix} a'_x & a \cos (C_L + \dot{C}_L t) \cos (C_N + \dot{C}_N t) & -a'_y \\ a'_y & a \sin (C_L + \dot{C}_L t) \cos (C_N + \dot{C}_N t) & a'_x \\ a'_z & -a \sin (C_L + \dot{C}_L t) & 0 \end{bmatrix}$$

and for the orbit plane system

$$g' = \begin{bmatrix} a'_x & -a'_y & -a \sin (\gamma + \dot{\gamma} t) \cos (\delta + \dot{\delta} t) \\ a'_y & a'_x & -a \sin (\gamma + \dot{\gamma} t) \sin (\delta + \dot{\delta} I) \\ a'_z & 0 & a \cos (\gamma + \dot{\gamma} t) \end{bmatrix}$$

g' is then transformed to the inertial reference system by

$$g = A g'$$

When SIMSEP calls TRAJ, time-varying noise is added to the thrust controls and the noised controls are used to compute $\underline{a}$ and g.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
T	I	C	Trajectory time in seconds.
CMASS	I	C	Current spacecraft mass.
EXHVEL	I	C	Exhaust velocity (c), (Equivalenced to ENGINE(10). Thruster efficiency (η), (Equivalenced to ENGINE(11).
NTPHAS	I	C	Current thrust phase number.
WPØWER	O	C	Power available (P).
UREL	I	C	Heliocentric position vector.
URELM	I	C	Position magnitude array.
NØISED	I	C	Flag that causes EP to add noise to the controls.
THRUST	I	C	Matrix that contains a set of controls for each segment. (THRUST (i, NTPHAS)) where i is the desired information for the NTPHAS phase. i = 1, thrust policy i = 2, phase end time in seconds i = 3, thrust scale factor i = 4, C_N or δ ; dependent upon i = 1 i = 5, C_L or γ ; dependent upon i = 1 i = 6, $\dot{C}_N$ or $\dot{\delta}$; dependent upon i = 1

Variable	Input/ Output	Argument/ Common	Definition
			i = 7, $\dot{C}_L$ or $\dot{\gamma}$; dependent upon i = 1
GT	O	C	$g = \frac{\partial \underline{a}}{\partial \underline{u}}$
THRACC	O	C	$\underline{a}$
UTRUE			Position vector relative to the primary body.
VTRUE			Velocity vector relative to the primary body.

Local Variables:

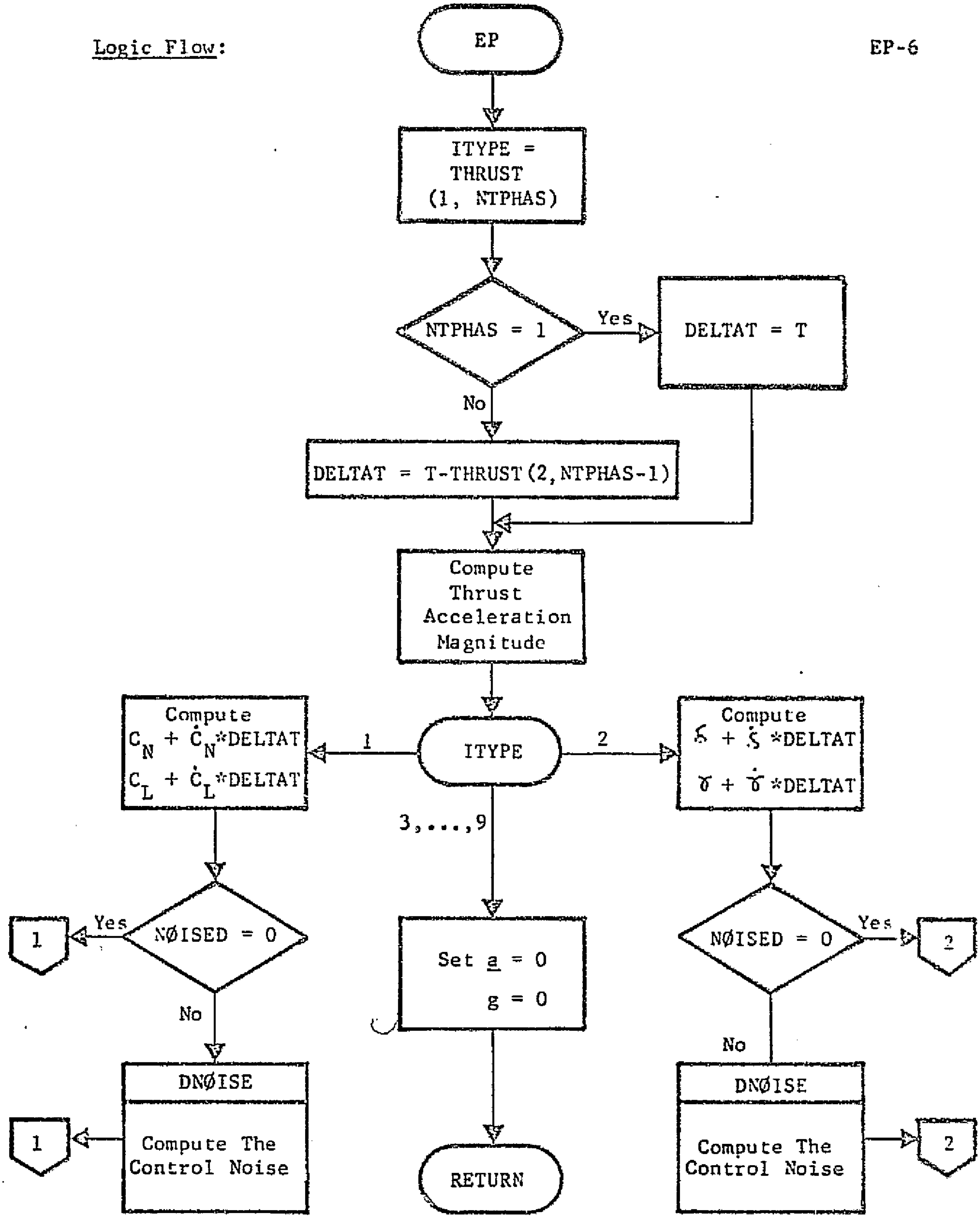
Variable	Definition
ACCEL	a
ACØNE	C_N
ACLØCK	C_L
AIN	δ
AØUT	γ
RØTMAT	The transformation matrix A.
WØRK	Used to store the matrix g′.
ITYPE	Thrust policy for the NTPHAS segment = THRUST (1, NTPHAS).
DELTAT	Time from the beginning of the control phase (t)

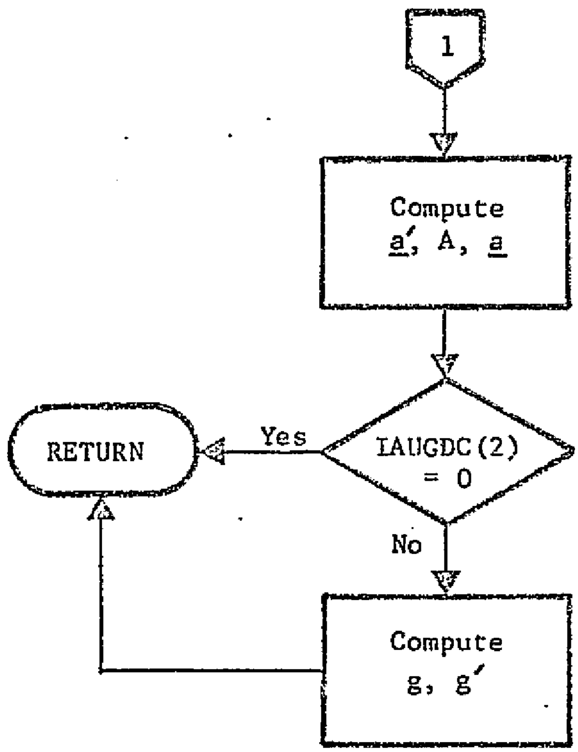
Subroutines Called: PØWER, DNØISE, UNITV, UXV, MMAB, ZERØM

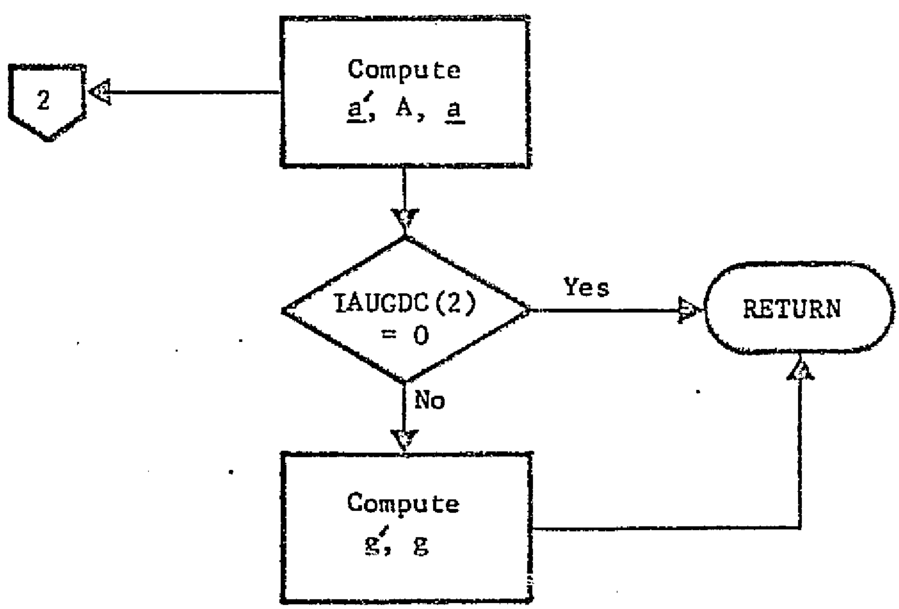
Calling Subroutines: MØTIØN

Common Blocks: CØNST, EPHEM, TRAJ1, TRAJ2, WØRK

Logic Flow:


EP
ITYPE = THRUST (1, NTPHAS)
NTPHAS = 1
Yes
DELTAT = T
No
DELTAT = T-THRUST(2,NTPHAS-1)
Compute Thrust Acceleration Magnitude
ITYPE
1
2
3,...,9
Compute $C_N + \dot{C}_N$*DELTAT $C_L + \dot{C}_L$*DELTAT
Compute $\delta + \dot{\delta}$ *DELTAT $\gamma + \dot{\gamma}$ *DELTAT
NØISED = 0
Yes
No
1
2
Set a = 0 g = 0
DNØISE
Compute The Control Noise
RETURN


1
Compute
a′, A, a
IAUGDC(2)
= 0
Yes
RETURN
No
Compute
g, g′


Compute
a′, A, a
2
IAUGDC(2)
= 0
Yes
RETURN
No
Compute
g′, g

3.5.4 Subroutine: EPHEM (NØ, DJ, R, V)

Purpose: To compute the heliocentric position and velocity vectors of a given planet or body.

Method: The orbital elements (a, e, i, Ω, $\tilde{\omega}$, M) of the desired body are computed from time varying expressions, for example, the semi-major axis

$$a(t) = a_o + a_1 t_J + a_2 t_J^2 + a_3 t_J^3$$

where a_o is the value at the ephemeris epoch 1900, January 0.5, t_J is the time from the epoch, and a_1, a_2, a_3 are constant coefficients. t_j is measured in days for all elements except mean anomaly of the planets where t_j is measured in units of 10^{-4} days. After the osculating orbital elements are computed, they are transformed into cartesian position and velocity vectors.

A unique case occurs when EPHEM is used to compute the position and velocity vectors of the earth's moon. The position ($\underline{r}_E$) and velocity ($\underline{V}_E$) vectors of the earth are computed and added to the position ($\underline{r}_M$) and velocity ($\underline{V}_M$) vectors of the moon relative to earth. The heliocentric position ($\underline{r}$) and velocity ($\underline{V}$) are

$$\underline{r} = \underline{r}_E + r_M$$

$$\underline{v} = \underline{v}_E + \underline{v}_M$$

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
NØ	I	A	Number of the planet for which $\underline{r}$ and $\underline{v}$ are desired.
DJ	I	A	Trajectory time in Julian Days from launch.
R	Ø	A	$\underline{r}$.
V	Ø	A	$\underline{v}$.
SMASS	I	C	Gravitational constant of the sun.
PMASS	I	C	Array of gravitational constants for the planets and the moon.
CSAX	I	C	Semi-major axis constants (a)
CESS	I	C	Eccentricity constants (e).
CINC	I	C	Inclination constants (i).
CØMEG	I	C	Longitude of the Ascending Node constants (Ω).
CØMEGT	I	C	Longitude of Periapsis constants ($\tilde{\omega}$).
CMEAN	I	C	Mean Anomaly constants (M).

Variable	Input/ Output	Argument/ Common	Definition
EMN	I	C	Array of constants for the moon. 1-4 Longitude of the Ascending Node constants. 5-8 Longitude of Periapsis constants. 9-12 Mean Anomaly constants. 13 Inclination constants. 14 Eccentricity constants. 15 Semi-major axis constants.
PI	I	C	3.14159.....(π)
DJ1900	I	C	2415020.

Local Variables:

Variable	Definition
XPLAN	Array used to store $\underline{r}_E$ and $\underline{v}_E$.
NP	Planet code, initially set equal to NØ.
PI2	$\frac{\pi}{2}$.
A	a.
E	e.
XI	i.
ØMEGA	Ω
SØMEGA	$\Omega - \tilde{\omega} = \omega$
XMEAN	M.

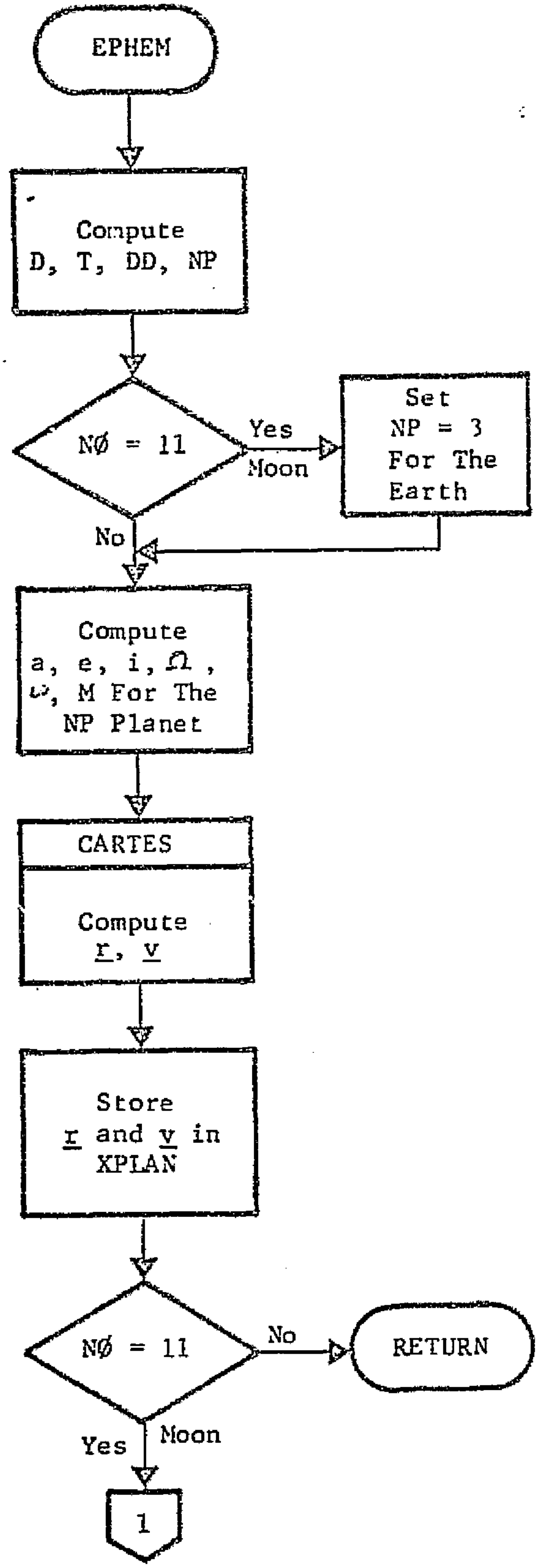
Variable	Definition
GMU	SMASS + PMASS(NP), for the planets. PMASS(3) + PMASS(11), for the moon.
PØLY3	Statement function that performs $\alpha_i(t) = a_i + t_J (b_i + t_J (c_i + d_i t_J))$
PØLY1	Statement function that performs $\alpha_i(t) = a_i + b_i t_J$
D	Days from 1900.
DD	D/10000.
T	D/36525.

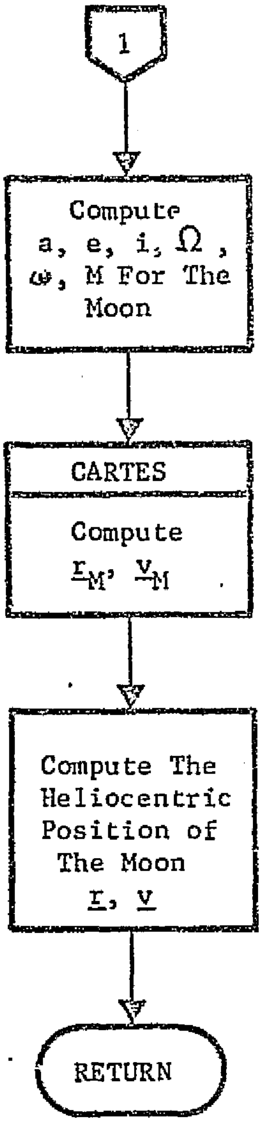
Subroutines Called: CAR[illegible]

Calling Subroutines: SØLAR

Common Blocks: CØNST, EPHEM

Logic Flow:


EPHEM
Compute D, T, DD, NP
NØ = 11
Yes Moon
Set NP = 3 For The Earth
No
Compute a, e, i, Ω, ω, M For The NP Planet
CARTES
Compute r, v
Store r and v in XPLAN
NØ = 11
No
RETURN
Yes Moon
1


1
Compute
a, e, i, Ω ,
ω, M For The
Moon
CARTES
Compute
$\underline{r}_M$, $\underline{v}_M$
Compute The
Heliocentric
Position of
The Moon
$\underline{r}$, $\underline{v}$
RETURN

C-6

3.5.5 Subroutine: FIND

Entry Points: FIND1, FIND3

Purpose: (1) To compute the location in Blank Common arrays that will be used by TRAJ and the number of equations to be integrated, (2) to copy integrated parameters into mode accessible locations, and (3) to initialize the F matrix.

Method: None

Remarks: All LØCXX variables indicate locations within Blank Common

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
LØCS	I	C	Location in Blank Common where TRAJ can start array allocation.
INTEG	I	C	Set = 1 Propagate State & Transition Matrix Set = 2 Propagate State only Set = 3 Propagate State and Covariance
IAUGDC	I	C	Flag array determining the components of the Transition Matrix or Covariance to be propagated.
MEQ	Ø	C	Total number of equations to be integrated.
MEQ8	Ø	C	MEQ-8
MEQS	Ø	C	$\sqrt{MEQ8}$
LØCH	Ø	C	Integration stepsize
LØCX	Ø	C	Trajectory time in seconds
LØCPT	Ø	C	Trajectory print time
LØCET	Ø	C	Trajectory event time
LØCPR	Ø	C	Trajectory time for print
LØCT	Ø	C	Trajectory time stored for interpolation

Input/Output:
(Continued)

Variable	Input/ Output	Argument/ Common	Definition
LØCR	Ø	C	Position magnitude stored for interpolation
LØCYC	Ø	C	Dependent variables
LØCDY	Ø	C	Differential equations
LØCYT	Ø	C	Dependent variables for print and events
LØCDT	Ø	C	Differential equations for print and events
LØCYP	Ø	C	Temporary locations for integration
LØCTE	Ø	C	Future modifications
LØCFI	Ø	C	F matrix, $\dot{\phi} = F\phi$
LØCM	Ø	C	Mass
LØCDM	Ø	C	Mass variation
LØCTC	Ø	C	Transition or Covariance matrix

Local Variables:

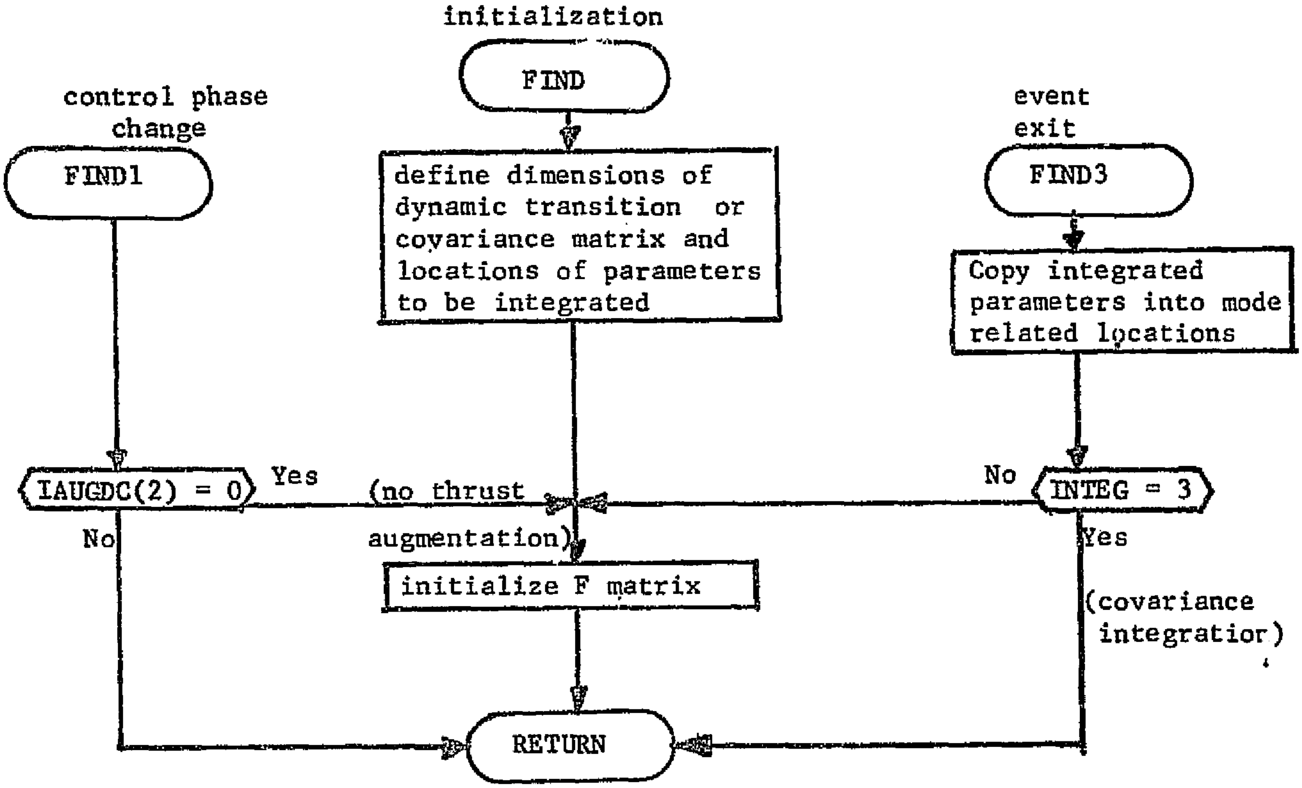
Variables	Definition
ISTATE	Array containing size of augmented dynamic parameters

Subroutines Called: CØPY, IDENT, MUNPAK, ZERØM

Calling Subroutines: PATH

Common Blocks: (BLANK), DIMENS, TRAJ1, TRAJ2, WØRK

Logic Flow:


initialization
FIND
control phase change
FIND1
event exit
FIND3
define dimensions of dynamic transition or covariance matrix and locations of parameters to be integrated
Copy integrated parameters into mode related locations
IAUGDC(2) = 0
Yes
No
(no thrust augmentation)
No
INTEG = 3
Yes
(covariance integratior)
initialize F matrix
RETURN

3.5.6-A Subroutine: GRAVAR

Purpose: GRAVAR computes the variational matrices, with the exception of the gravity gradient matrix (G11), needed to formulate the matrix differential equations which integrate into the augmented state transition matrix.

Method: The variational matrices are formulated as follows (Reference 1, p 122):

$$G12 = k = \frac{\partial \ddot{\underline{r}}}{\partial \underline{r}_e} = \frac{\mu_e}{r_e^5}\left[3\underline{r}_e\underline{r}_e^T - r_e^2 \; I\right] - \frac{\mu_s}{\rho_e^5}\left[3\underline{\rho}_e\underline{\rho}_e^T - \rho_e^2 \; I\right]$$

$$G22 = p = \frac{\partial \ddot{\underline{r}}_e}{\partial \underline{r}_e} = -\frac{\mu_e}{r_e^5}\left[3\underline{r}_e\underline{r}_e^T - r_e^2 \; I\right]$$

$$GM11 = m = \frac{\partial \ddot{\underline{r}}}{\partial \mu_s} = -\frac{\underline{r}}{r^3}$$

$$GM12 = d = \frac{\partial \ddot{\underline{\rho}}_e}{\partial \mu_e} = -\frac{\underline{\rho}_e}{\rho_e^3}$$

$$GM21 = s = \frac{\partial \ddot{\underline{r}}_e}{\partial \mu_s} = -\frac{\ddot{\underline{r}}_e}{r_e^3}$$

$$GM22 = q = \frac{\partial \ddot{\underline{r}}_e}{\partial \mu_e} = -\frac{\ddot{\underline{r}}_e}{r_e^3}$$

where:

$\underline{r}$ is the s/c heliocentric position vector

$\underline{r}_e$ is the heliocentric ephemeris planet position vector

μ_e is the gravitational constant of the ephemeris planet

μ_s is the gravitational constant of the sun

$\underline{\rho}_e$ is the position vector of the s/c WRT the ephemeris planet

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
UP	I	C	Heliocentric position vectors of all bodies in the integration
IAUGDC	I	C	Array of flags used to augment the state for STM or covariance integration
PMASS	I	C	Planetary gravitational constants
SMASS	I	C	Solar gravitational constant
UREL	I	C	Position vector of s/c relative to all bodies considered in the integration
URELM	I	C	Magnitudes of UREL
G12	Ø	C	k
G22	Ø	C	p
GM11	Ø	C	m
GM12	Ø	C	d
GM21	Ø	C	s
GM22	Ø	C	q
IEP	I	C	Ephemeris body identification

Local Variables:

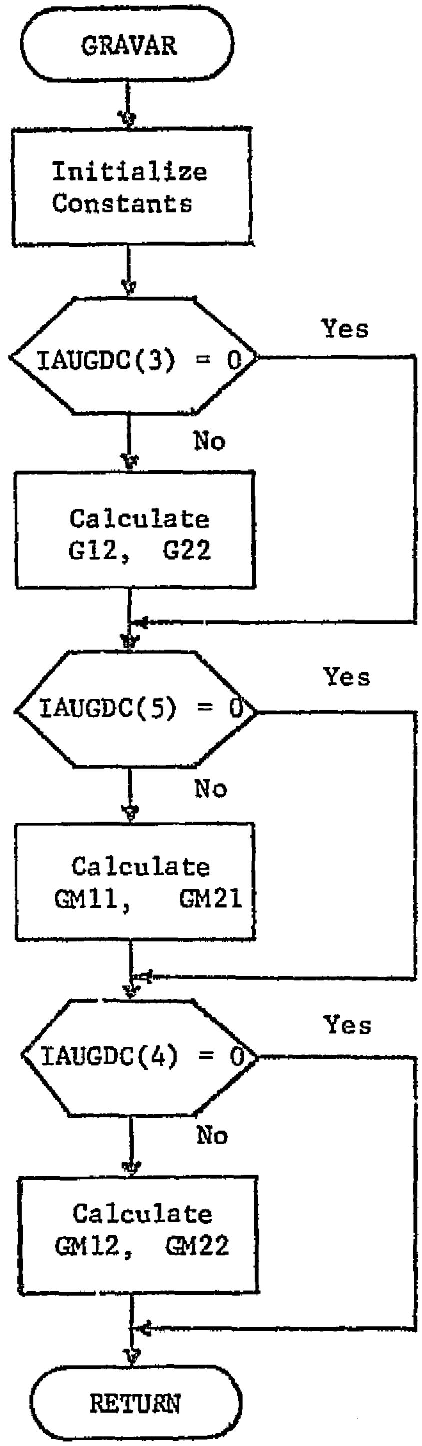
Variable	Definition
UPM (= WORK(10))	Magnitude of position vector of the ephemeris planet.
SMUK (= WORK(4))	Gravitational constant of ephemeris planet

Subroutines Called: VECMAG

Calling Subroutines: DPHI, PDOT

Common Blocks: EPHEM, TRAJ1, TRAJ2, WORK

Logic Flow:


GRAVAR
Initialize Constants
IAUGDC(3) = 0
Yes
No
Calculate G12, G22
IAUGDC(5) = 0
Yes
No
Calculate GM11, GM21
IAUGDC(4) = 0
Yes
No
Calculate GM12, GM22
RETURN

3.5.6-B **Subroutine:** GRAVFØ (UA)

Purpose: The subroutine GRAVFØ has two principal purposes. The first is the calculation of differential accelerations acting on the s/c due to gravitational bodies being considered in the analysis. The second purpose is the computation of the gravity gradient matrix, G11, which is used in the algorithm determining the step size for the trajectory integrator (PATH). G11 is used also with the other variational matrices, G12, G22, GM12, GM12, GM21, and GM22 (all computed in GRAVAR) to formulate the matrix differential equations which integrate into the augmented state transition matrix. In addition, GRAVFØ performs many auxilliary calculations which determine the relative geometrics among all planetary bodies and the s/c. These geometricl quantities are stored in common blocks accessible to other routines where they may be used without further computational expense.

Method: TRAJ uses Encke's formulation of the equations of motion for propagating trajectories, (Section 4.1, Reference 1). The differential acceleration computed by GRAVFØ is

$$\delta \ddot{\underline{r}} = -\frac{\mu}{r_c^{\,3}} \left[f(\alpha) \cdot \underline{r} + \delta \underline{r} \right] - \sum_{i=1}^{N} \frac{\mu}{\rho_i^3} \cdot$$

$$\left[\underline{r} + f(\alpha_i) \cdot \underline{r}_i \right]$$

where

$$\underline{r} = \underline{r}_c + \delta \underline{r}$$

$$\dot{\underline{r}} = \dot{\underline{r}}_c + \delta \dot{\underline{r}}$$

$$f(\alpha) = \frac{\alpha\,(3 + 3\alpha + \alpha^2)}{1 + (1 + \alpha)^{3/2}}$$

$$\alpha = \frac{(\delta \underline{r} - 2\underline{r}) \cdot \delta \underline{r}}{r^2}$$

$$\underline{\rho}_i = \underline{r} + \underline{r}_p - \underline{r}_i$$

$$f(\alpha_i) = \alpha_i \left[\frac{3 + 3\alpha_i + \alpha_i^2}{1 + (1 + \;_i)^{3/2}} \right]$$

$$\alpha_i = \frac{r}{\rho_i} \left[\frac{r}{\rho_i} - \frac{2\,\underline{r} \cdot \underline{\rho}_i}{r\,\rho_i} \right]$$

$\underline{r}_c$ - reference conic position vector of the spacecraft.

$\underline{\rho}_i$ - position vector of the spacecraft relative to the i^{th} body.

$\underline{r}$ - heliocentric position vector of the spacecraft.

$\underline{r}_i$ - heliocentric position vector of the i^{th} body.

N - number of bodies included in the integration other than the sun.

$\underline{r}_p$ - heliocentric position vector of the primary body.

μ - gravitational constant.

GRAVFØ also computes the gravity gradient matrix, G11, which is used for state transition matrix propagation and as a determinant in the integrator step size logic. (Reference 1, p 122)

$$G11 = f = \left(\sum_{i=1}^{N} f_i \right) + f_p$$

$$= \left(\sum_{i=1}^{N} \frac{\partial \underline{\ddot{\rho}}_i}{\partial \underline{\rho}_i} \right) + \frac{\partial \underline{\ddot{\rho}}_p}{\partial \underline{\rho}_p}$$

$$= \left(\sum_{i=1}^{N} \frac{\mu_i}{\rho_i^5} \left[3 \underline{\rho}_i \underline{\rho}_i^T - \underline{\rho}_i^2 I \right] \right)$$

$$+ \frac{\mu_p}{\rho_p^5} \left[3 \underline{\rho}_p \underline{\rho}_p^T - \underline{\rho}_p^2 I \right]$$

The subscript i refers to the i^{th} perturbing body and the subscript p refers to the primary body. $\underline{\rho}$ indicates body relative position vectors while μ is the gravitational constant.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
UA	I	A	The first three elements contain $\delta \underline{r}$. The second three elements contain $\delta \dot{\underline{r}}$.
UENC	I	C	$\underline{r}_c$
UENCM	I	C	r_c
VENC	I	C	$\dot{\underline{r}}_c$
UTRUE	O	C	$\underline{r}$
UTRUEM	O	C	r
VTRUE	O	C	$\dot{\underline{r}}$
VTRUEM	O	C	$\dot{r}$
APERT	O	C	Array that contains the perturbing acceleration vector for each body included in the integration. APERT (I, IPRI) I = 1.3 contains the vector sum of these perturbations.
SMASS	I	C	Solar gravitational constant.
PMASS	I	C	Array of planetary gravitational constants.
UREL	O	C	Array containing each $\underline{\rho}_i$.
URELM	O	C	Array containing each ρ_i.
VREL	O	C	Array containing each $\dot{\underline{\rho}}_i$.
VRELM	O	C	Array containing each $\dot{\rho}_i$.
UP	I	C	Array containing each $\underline{r}_i$.
VP	I	C	Array containing each $\dot{\underline{r}}_i$.

Variable	Input/ Output	Argument/ Common	Definition
NB	I	C	Array containing planet codes of each body in the integration.
APRIM	O	C	$\ddot{\underline{r}}_p$
ATØT	O	C	$\delta\ddot{\underline{r}}$
G11	O	C	f
MPLAN	I	C	N + 1
IPRI	I	C	Flag used to locate information concerning the primary body in the UP, UREL, URELM, VP, VREL, and VRELM arrays.

Local Variables:

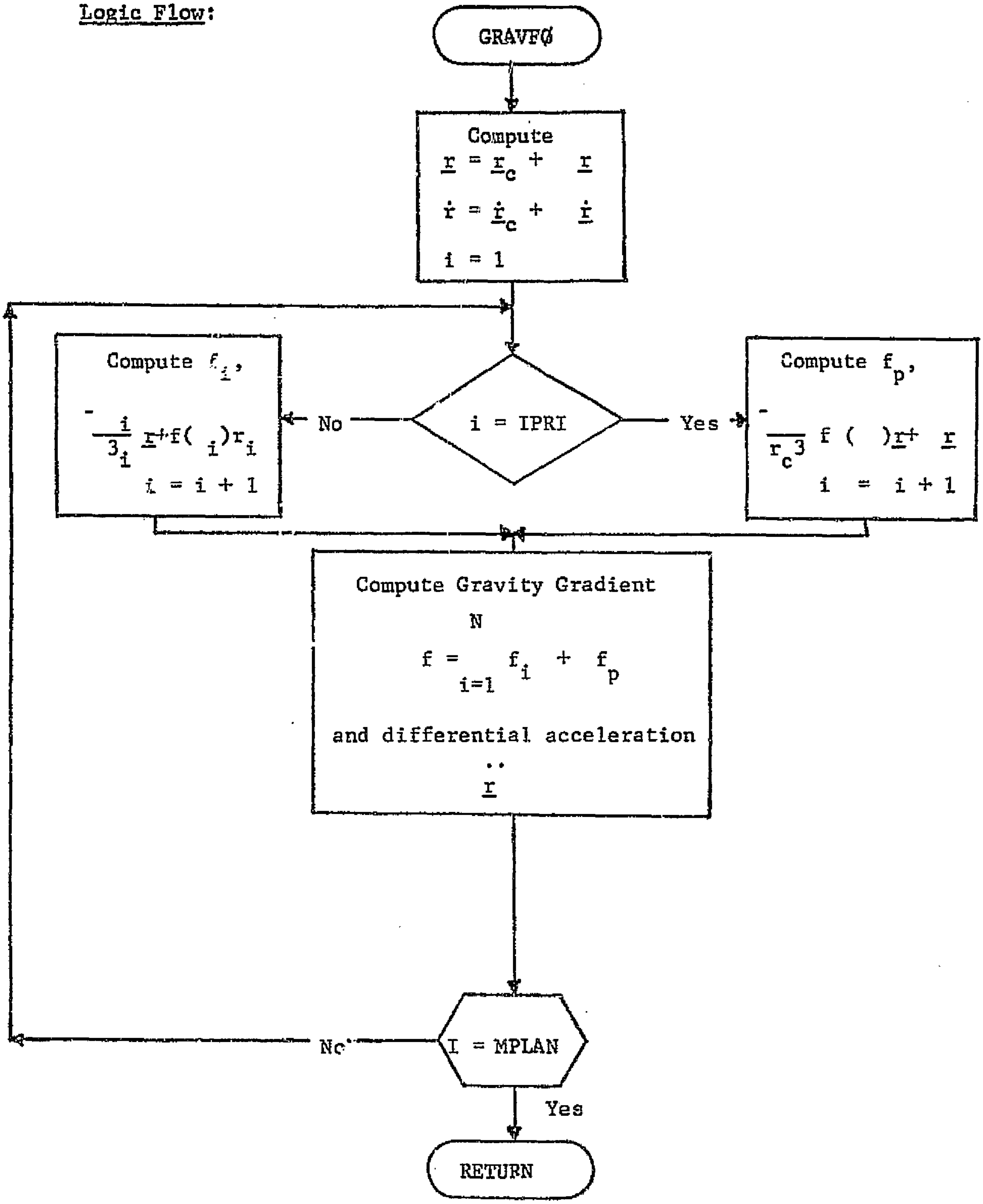
Variable	Definition
ADEL (= WØRK(I), I = 1,3)	$-\frac{\mu}{r_c^3}\left[f(\alpha)\,\underline{r} + \delta\underline{r}\right]$
APERT (J, IPRI), J = 1,3	$-\sum_{i-1}^{N}\frac{\mu_i}{\rho_i^3}\left[\underline{r} + f(\alpha_i)\,\underline{r}_i\right]$
F(X)	Statement function equivalent to $f(\alpha)$ and $f(\alpha_i)$.
Q (= WORK(21))	α

Subroutines Called: VECMAG

Calling Subroutines: MØTIØN

Common Blocks: EPHEM, TRAJ1, TRAJ2, WØRK

Logic Flow:


GRAVFØ
Compute
r = r_c + r
ṙ = ṙ_c + ṙ
i = 1
i = IPRI
No
Yes
Compute f_i,
i = i + 1
Compute f_p,
i = i + 1
Compute Gravity Gradient
N
f = f_i + f_p
i=1
and differential acceleration
r̈
I = MPLAN
No
Yes
RETURN

3.5.7 Subroutine: LØADFM (DS, DP, INDEX)

Purpose: To compute the F matrix and the matrix of derivatives $\dot{\Phi} = F\Phi$ or $\dot{P} = FP + PF^T + Q$ for transition matrix or covariance, respectively. (Sections 4.5 and 4.6, Reference 1).

Method: The non-zero components of F are stored in appropriate sub-matrices, according to the degree the state is augmented.

Remarks: Case 1: State transition matrix.

Given the augmented state vector

$$\underline{x} = \begin{bmatrix} \underline{r} \\ \dot{\underline{r}} \\ \underline{u} \\ \underline{r}_e \\ \dot{\underline{r}}_e \\ \mu_e \\ \mu_s \end{bmatrix}$$

where

- r - spacecraft position vector.
- $\dot{\underline{r}}$ - spacecraft velocity vector.
- $\underline{u}$ - constant spacecraft controls.
- $\underline{r}_e$ - position vector of the spacecraft relative to the ephemeris body.
- $\dot{\underline{r}}_e$ - velocity vector of the spacecraft relative to the ephemeris body.

The linearized equations of motion for the augmented state are

$$\delta \dot{\underline{x}} = F \, \delta \underline{x}$$

where

$$F = \frac{\partial \dot{\underline{x}}}{\partial \underline{x}}$$

$$F = \begin{bmatrix} 0 & I & 0 & 0 & 0 & 0 & 0 \\ f & 0 & g & k & 0 & d & m \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & I & 0 & 0 \\ 0 & 0 & 0 & p & 0 & q & s \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

where I is a 3x3 identity matrix and

$$f = \frac{\partial \ddot{\underline{r}}}{\partial \underline{r}} \qquad m = \frac{\partial \ddot{\underline{r}}}{\partial \mu_s}$$

$$g = \frac{\partial \ddot{\underline{r}}}{\partial \underline{u}} \qquad p = \frac{\partial \ddot{\underline{r}}_e}{\partial \underline{r}_e}$$

$$k = \frac{\partial \ddot{\underline{r}}}{\partial \underline{r}_e} \qquad q = \frac{\partial \ddot{\underline{r}}_e}{\partial \mu_e}$$

$$d = \frac{\partial \ddot{\underline{r}}}{\partial \mu_e} \qquad s = \frac{\partial \ddot{\underline{r}}_e}{\partial \mu_s}$$

<u>Case 2</u>: Covariance matrix.

Given the augmented state vector

$$\underline{x} = \begin{bmatrix} \underline{r} \\ \underline{r} \\ \underline{u} \\ \underline{\omega} \\ \underline{r}_1 \\ \underline{r}_2 \\ \underline{r}_3 \end{bmatrix}$$

where

$\underline{\omega}$ - time varying thrust parameters.

$\underline{r}_i$ - tracking station position vectors.

and

$$F = \begin{bmatrix} 0 & I & 0 & 0 & 0 & 0 & 0 \\ f & 0 & g & n & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & h & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

where I is a 3x3 identity matrix,

$$f = \frac{\partial \underline{\ddot{r}}}{\partial \underline{r}}$$

$$g = \frac{\partial \ddot{\underline{r}}}{\partial \underline{u}}$$

$$n = [g \vdots g]$$

and h is the matrix of process noise correlation times

$$h = \begin{bmatrix} \frac{-1}{\tau_1} & 0 & ---- & 0 \\ 0 & \frac{-1}{\tau_2} & ---- & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & ---- & \frac{-1}{\tau_6} \end{bmatrix} .$$

The matrix Q is the process noise,

$$Q = \begin{bmatrix} 0 & 0 & 0 & & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -2 \cdot h \cdot E[\delta \underline{\omega}\, \delta \underline{\omega}^T] & & 0 & 0 & 0 \\ 0 & 0 & 0 & & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & & 0 & 0 & 0 & 0 \end{bmatrix}$$

The dimensions of $\underline{\Phi}$, $\dot{\underline{\Phi}}$, P, $\dot{P}$, F and Q are determined by the highest degree of augmentation of the state vector. The flag array that controls the augmentation is the IAUGDC array.

Input/Output:

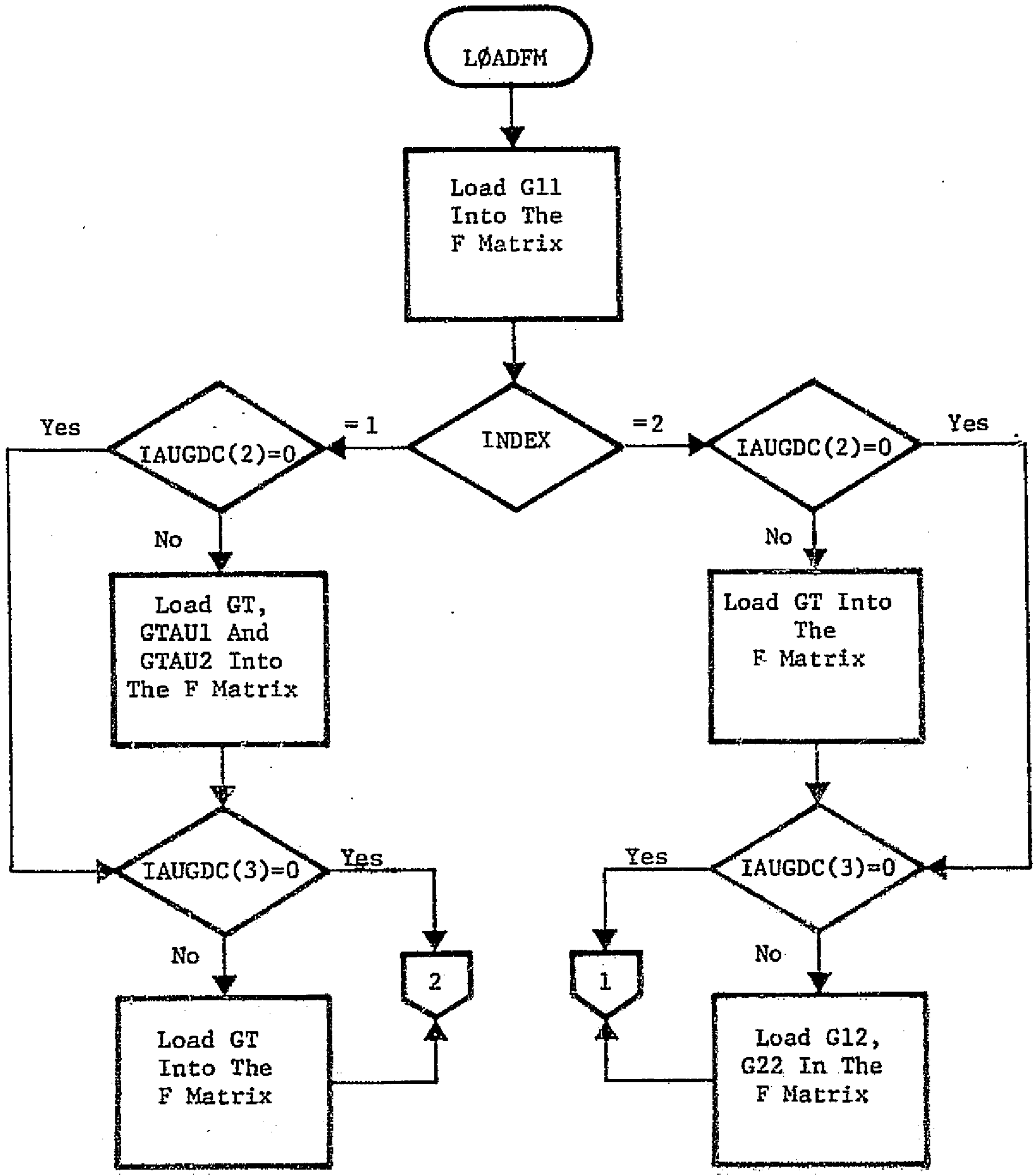
Variable	Input/ Output	Argument/ Common	Definition
INDEX	I	A	= 1, Load the F matrix and compute $\dot{P}$. = 2, Load the F matrix and compute $\dot{\Phi}$. = 3, Use current F, compute $\dot{P}$. = 4, Use current F, compute $\dot{\Phi}$.
DS	I	A	= P for Covariance propagation. = Φ for Transition Matrix propagation
DP	O	A	= $\dot{P}$ for Covariance propagation, $\dot{\Phi}$ for transition matrix
F(LØCFI)	I	C	Location in Blank Common to use for F matrix storage.
IAUGBC	I	C	Array of flags where each element determines what is to be loaded in the F matrix.
G11	I	C	f
GT	I	C	g
G12	I	C	k
G22	I	C	p
GM12	I	C	d
GM22	I	C	q
GM11	I	C	m
GM21	I	C	s
GTAU1	I	C	Upper left 3x3 of h
GTAU2	I	C	Lower right 3x3 of h
QNØISE	I	C	Q = process noise
MEQS	I	C	Dimensions of Φ, $\dot{\Phi}$, P, $\dot{P}$, and F.

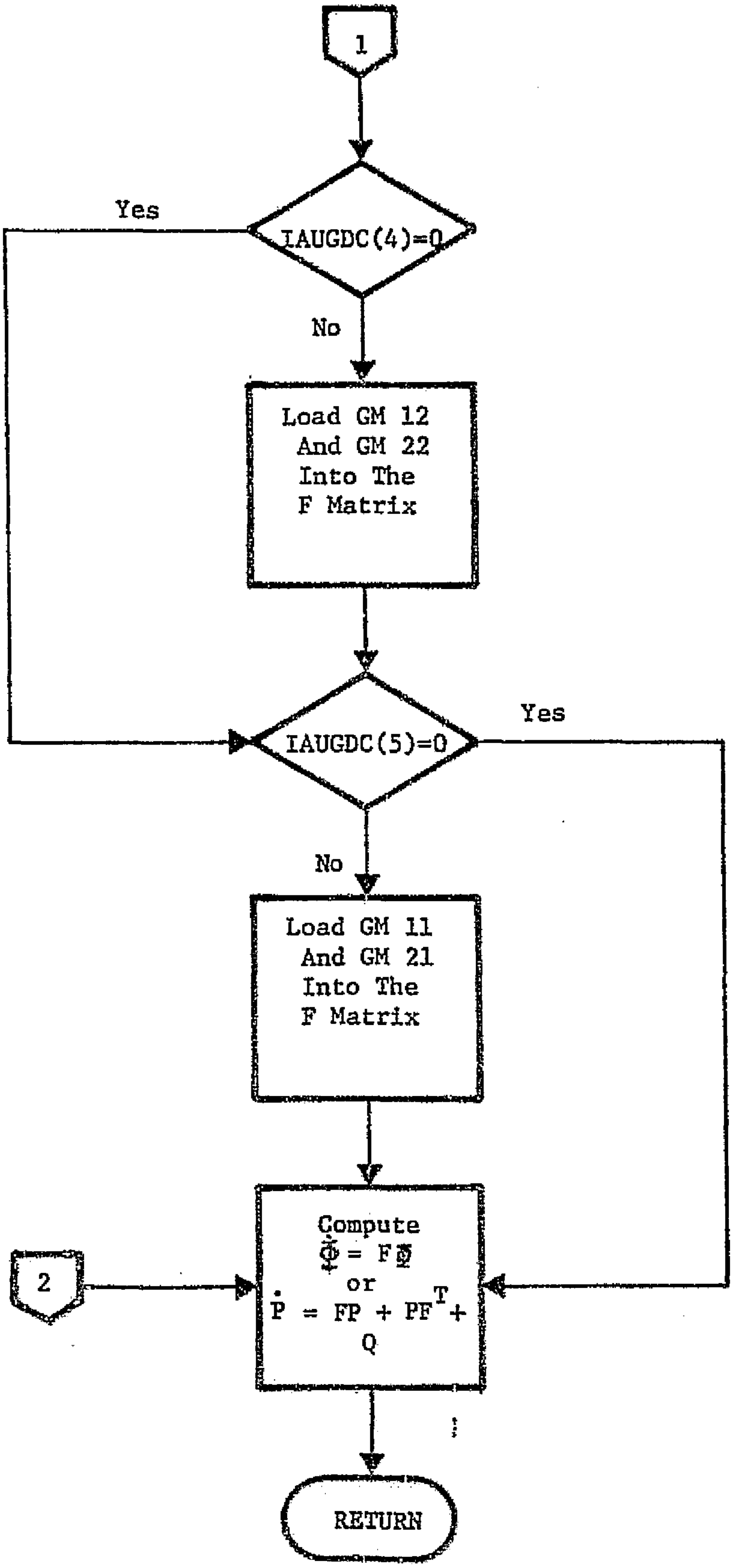
Subroutines Called: MMAB, MUNPAK, SCALE, SYMTRZ, ZERØM

Calling Subroutines: DPHI, PDØT, TRAJ

Common Blocks: (BLANK), TRAJ1, TRAJ2, WØRK

Logic Flow:




1
IAUGDC(4)=0
Yes
No
Load GM 12 And GM 22 Into The F Matrix
IAUGDC(5)=0
Yes
No
Load GM 11 And GM 21 Into The F Matrix
2
Compute $\dot{\Phi} = F\Phi$ or $\dot{P} = FP + PF^T + Q$
RETURN

3.5.8 Function Routine: LØCATE (INDEX)

Purpose: To locate the target body, ephemeris body, launch body or primary body in the NB array.

Method: None

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
INDEX	I	A	SET = 1 Locate target body = 2 Locate ephemeris body = 3 Locate launch body = 4 Locate primary body
NTP	I	C	Number of the target body
NEP	I	C	Number of the ephemeris body
NLP	I	C	Number of the launch body
NPRI	I	C	Number of the primary body

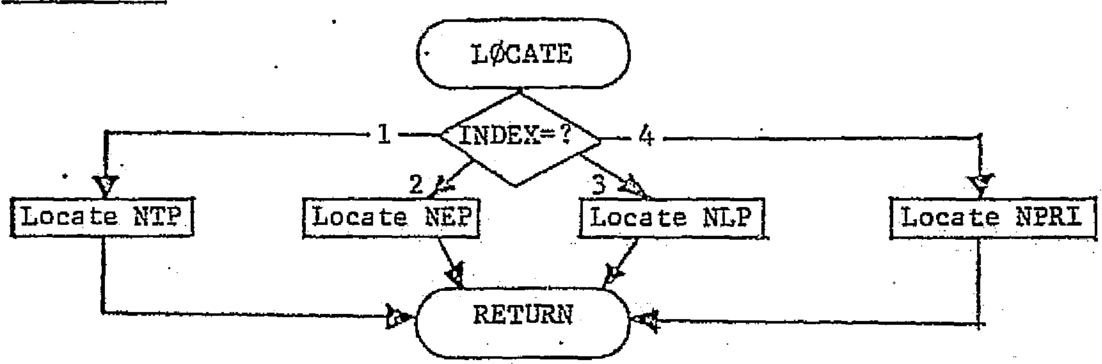
Local Variables: None

Subroutines Called: None

Calling Subroutines: PATH, GRAVFØ

Common Blocks: TRAJ2

Logic Flow:



3.5.9 Subroutine: MØTIØN (T, DS, DSD, M, N, LØC)

Purpose: To compute the S/C accelerations and to rectify the reference conic.

Method: Encke's formulation of the equations of motion.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
T	I	A	Trajectory time
DS	I	A	Dependent variable
DSD	Ø	A	Differential equations
M	I	A	Number of rows in DS and DSD
N	I	A	Number of columns in DS and DSD
LØC	I	A	Routing flag
EPØCH	I	C	Julian Date of Launch
TM	I	C	Conversion from seconds to days
EXHVEL	I	C	Exhaust velocity
ATØT	I	C	Differential acceleration plus perturbing gravitational accelerations
THRACC	I	C	Thrust accelerations
RPACC	I	C	Radiation Pressure acceleration

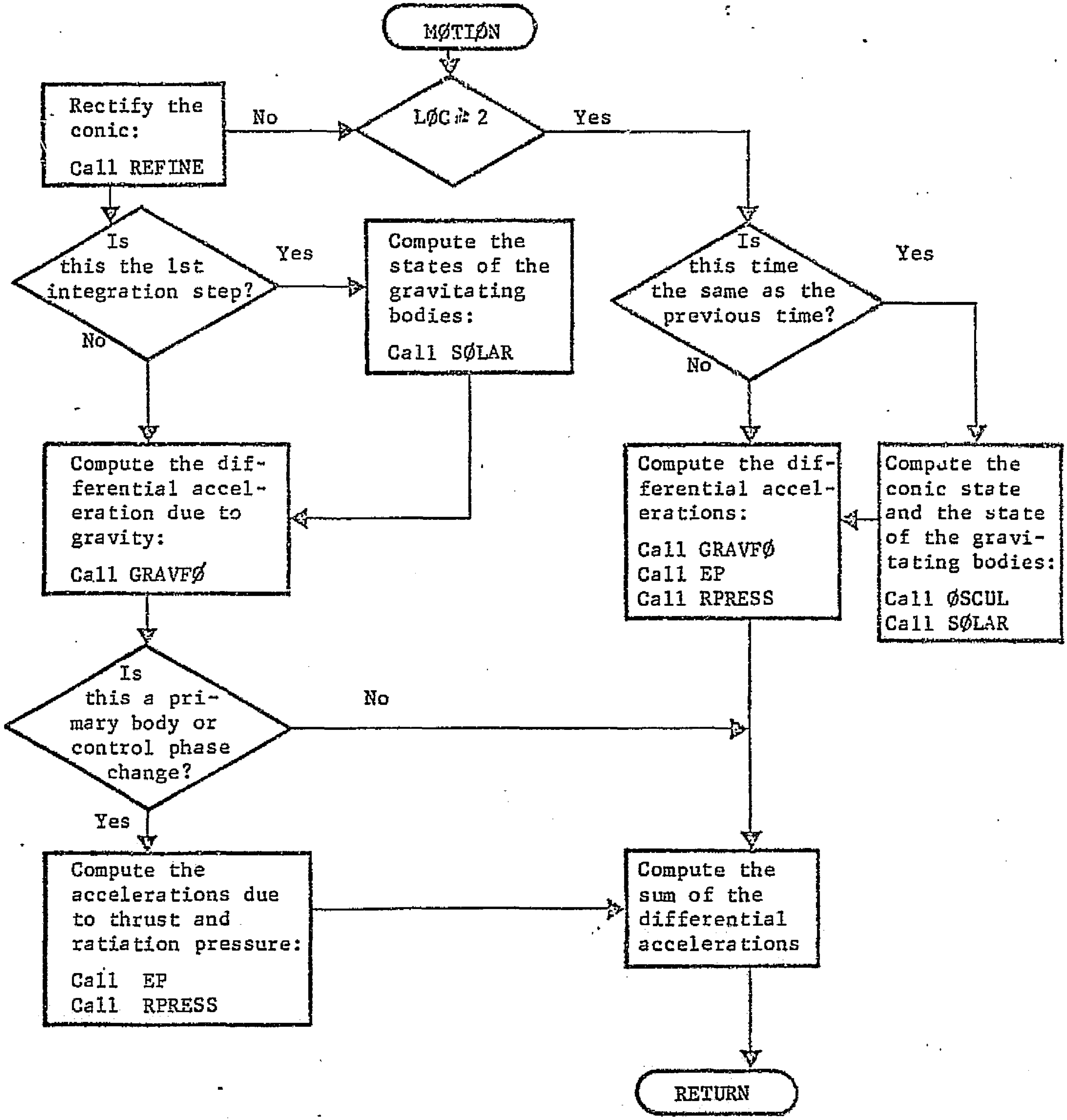
Local Variables: None

Subroutines Called: REFINE, SØLAR, ØSCUL, GRAVFØ, EP, RPRESS, ADD, CØPY

Calling Subroutines: NUMIN, DPHI, PDØT

Common Blocks: CØNST, TIME, TRAJ1, TRAJ2, WØRK

Logic Flow:


MØTIØN
LØC ≠ 2
No
Yes
Rectify the conic:
Call REFINE
Is this the 1st integration step?
Yes
No
Compute the states of the gravitating bodies:
Call SØLAR
Is this time the same as the previous time?
Yes
No
Compute the differential acceleration due to gravity:
Call GRAVFØ
Compute the differential accelerations:
Call GRAVFØ
Call EP
Call RPRESS
Compute the conic state and the state of the gravitating bodies:
Call ØSCUL
Call SØLAR
Is this a primary body or control phase change?
No
Yes
Compute the accelerations due to thrust and ratiation pressure:
Call EP
Call RPRESS
Compute the sum of the differential accelerations
RETURN

3.5.10 Subroutine: NEWTØN (XVALUE, YVALUE, X, Y, INDEX)

Purpose: To fit a third Order Polynomial through 4 data points for either interpolation or finding the minimum of the polynomial.

Method: Newton's third Order Divided Difference Interpolation Polynomial. (See Appendix 3, Reference 1)

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
XVALUE	I	A	Table of independent values
YVALUE	I	A	Table of dependent values
X	I/O	A	For interpolation, the value of X for which Y is desired. (Input) For a minimum, the value of X at the minimum. (Output)
Y	I/O	A	For interpolation, the interpolated value of Y. (Output) For a minimum, the value of Y at the minimum. (Output)
INDEX	I	A	Set = 1, Find the minimum Set = 2, Interpolate

Local Variables:

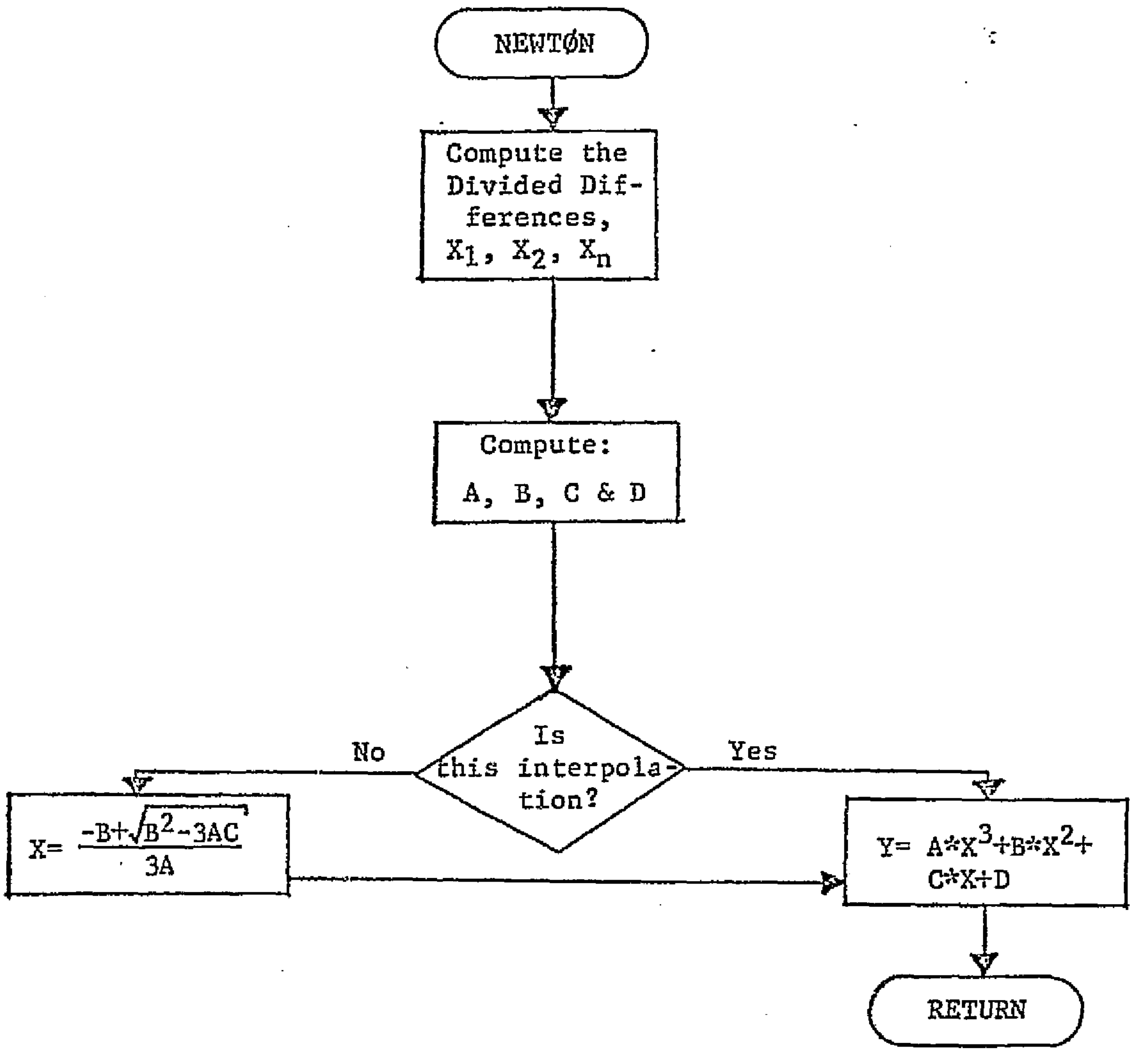
Variable	Definition
DDX	The Divided Differences
A, B, C, D	Coefficients of a 3rd Order Polynomial

Subroutines Called: None

Calling Subroutines: PATH

Common Blocks: None

Logic Flow:


NEWTØN
Compute the Divided Differences, X1, X2, Xn
Compute:
A, B, C & D
Is this interpolation?
No
Yes
X= (-B+√(B²-3AC))/3A
Y= A*X³+B*X²+C*X+D
RETURN

3.5.11 Subroutine: NUMIN (M, N, X, H, YC, YP, F, DERIV)

Entry Points: SETUP, RUNG2, RUNG4

Purpose: To integrate an MxN matrix of first order differential equations.

Method: 4th Order Runge-Kutta formula (RUNG4) and 2nd Order (RUNG2)

Input/Output:

Variable	I/O	Argument/ Common	Definition
M	I	A	Number of rows
N	I	A	Number of columns
X	I/Ø	A	Independent variable
H	I	A	Integration step-size
YC	I/Ø	A	Matrix of dependent variables
YP	Ø	A	Temporary storage matrix
F	Ø	A	4 - Temporary storage matrices
DERIV	I	A	Name of the subroutine containing the differential equations.

Local Variables:

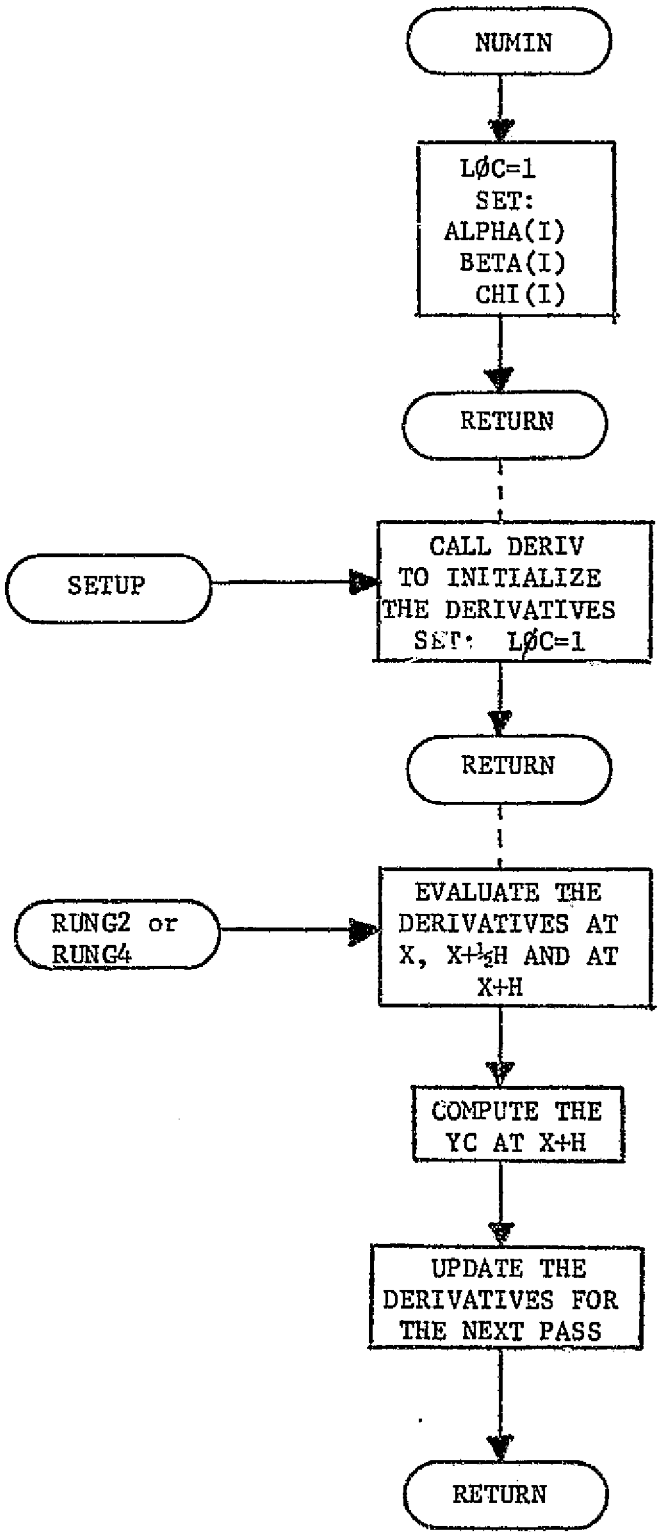
Variable	Definition
ALPHA	Array of 4 integration constants (0, ½, ½, 1) or (0, 1, 0, 0)
BETA	Array of 4 integration constants (0, ½, ½, 1) or (0, 1, 0, 0)
CHI	Array of 4 integration constants (1/6, 2/6, 2/6, 1/6) or (½, ½, 0, 0)
LØC	Output flag to DERIV

Subroutines Called: DERIV (defined by argument, e.g., DPHI, MØTIØN, PDØT)

Calling Subroutine: PATH

Common Blocks: None

Logic Flow:


NUMIN
LØC=1
SET:
ALPHA(I)
BETA(I)
CHI(I)
RETURN
SETUP
CALL DERIV TO INITIALIZE THE DERIVATIVES
SET: LØC=1
RETURN
RUNG2 or RUNG4
EVALUATE THE DERIVATIVES AT X, X+½H AND AT X+H
COMPUTE THE YC AT X+H
UPDATE THE DERIVATIVES FOR THE NEXT PASS
RETURN

3.5.12 Subroutine: PATH

Entry Point: FLIGHT

Purpose: PATH initializes all trajectory routines, while FLIGHT controls trajectory propagation.

Remarks: Based upon input flags, PATH determines how FLIGHT will function as well as all the other trajectory routines. FLIGHT tests for and executes trajectory rectification, primary body changes, thrust control changes, trajectory termination conditions, trajectory print and trajectory events.

The most significant feature of PATH is the use of blank common as a working area for the Fourth Order Runge-Kutta numerical integration routine (Appendix 2, Reference 1), applied to a matrix of first order differential equations.

$$Y_{k+1} = Y_k + \frac{h_k}{6} \quad (F_1 + 2 \cdot F_2 + 2 \cdot F_3 + F_4)$$

where

$$F_1 = F'(x_k, Y_k)$$

$$F_2 = F'(x_k + \frac{h_k}{2}, Y_k + \frac{h_k}{2} \cdot F_1)$$

$$F_3 = F'(x_k + \frac{h_k}{2}, Y_k + \frac{h_k}{2} \cdot F_2)$$

$$F_4 = F'(x_k + h_k, Y_k + h_k \cdot F_3)$$

The values of Y and F are stored in a blank common array (DSC) and their order depends upon whether some or no events are processed within the normal integration step (h_k).

Case 1: If no events occur between X_k and $X_{k+1} = X_k + h_k$, then a normal integration step will be taken. The values of Y_k and F_1 (X_k, Y_k) are used for the Runge-Kutta integration and at the completion of the step the DSC array appears as

$$DSC = Y_{k+1}, F_1(X_k + h_k, Y_{k+1}),$$

$$F_2, F_3, F_4, \_\_\_, \_\_\_, Y_{k+1}$$

where the first two entries (Y and F_1) are at the updated X_{k+1} point, the next three entries contain values of F in the h_k interval, there are two unused storage arrays, and the last entry is a running value of Y (which becomes Y_{k+1} at the end of the step). The next integration step (h_{k+1}) can now be taken and starts with Y_{k+1}, F_1.

Case 2: If an event or print has been specified by either the calling mode or TRAJ itself, and it occurs between X_k and X_{k+1}, then a short integration step (*h) is taken to the event. The resultant blank common storage at the event (X_k + *h) is then

$$DSC = Y_k, F_1, {}^*Y_{k+1}, {}^*F_1(x_k + {}^*h_k, {}^*Y_{k+1}),$$

$${}^*F_2, {}^*F_3, {}^*F_4, {}^*Y_{k+1}$$

where asterisks (*) refer to values for the event integration step. The first two entries are stored values of Y and F at X_k, to preserve values such that a normal integration step can be taken after the event has been processed. The next six entries are used for the event integration step. If no more events occur before $X_k + h_k$, then normal integration resumes with the stored values Y_k and F_1, and the results are shown in Case 1. If more events occur before $X_k + h_k$, then the process of Case 2 is repeated using $^*Y_{k+1}$ and $^*F_1$ until all events have been processed. Since TRAJ can integrate the transition matrix or covariance in addition to the state deviation from the reference conic, an additional array is needed. This array is used to store the partial deviatives contained in the F matrix (Appendix 4, Reference 1). The locations for the F matrix begin after the last word of Y_k (or $*Y_k$). The amount of blank common used by TRAJ varies with the number of equations to be integrated. For the state only case,

$$Y = \begin{bmatrix} \delta \underline{r} \\ \delta \underline{v} \\ m \\ \delta m \end{bmatrix}$$

where $\delta \underline{r}$ and $\delta \underline{v}$ are deviations from the conic state, m is the spacecraft mass and δm is the mass variation. When the transition matrix (Φ) or the covariance (P) are to be integrated

$$Y = \begin{bmatrix} \delta \underline{r} \\ \delta \underline{v} \\ m \\ \delta m \\ \underline{\Phi} \end{bmatrix}$$

or

$$Y = \begin{bmatrix} \delta \underline{r} \\ \delta \underline{v} \\ m \\ \delta m \\ P \end{bmatrix}$$

For state only integration, Y is an 8 x 1 matrix. When the transition matrix or covariance is to be integrated, the dimension of Y varies with $\tilde{\Phi}$ and P. The dimensions of $\tilde{\Phi}$ and P are those for the highest degree of augmentation. The subroutine FIND determines the number of equations to be integrated, the dimensions of $\tilde{\Phi}$ or P and the number of locations in blank common needed for numerical integration.

Other information stored in blank common are:

- X_k - Current trajectory integration time (t);
- h - Integration stepsize;
- t_p - Integration event time;
- t_e - Next mode event time;
- t_{PR} - Next mode print time;
- t_i - Four stored times used for interpolation;
- r_i - Four stored position magnitudes corresponding to the t_i's, also used for interpolation;

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
INTEG	I	C	Flag that determines the equations to be integrated.

Variable	Input/ Output	Argument/ Common	Definition
			= 1, State and transition matrix; = 2, State; = 3, State and covariance.
IPRINT	I	C	Flag that determines when to print. = 1, Every IPRINT integration step; = 0, No print; =-1, Every XPRINT days; =-2, At trajectory event.
MPLAN	I	C	Total number of bodies to be considered in the NB array.
LØCS	I	C	First location the integration routine can use for storage.
ISTØP	I	C	Flag that determines trajectory termination. 1 - Final trajectory time (TDUR); 2 - Radius of Closest Approach to the target body; 3 - Sphere of influence of the target body; 4 - Stopping radius relative to the target body.
KTRAJ	I	C	Flag used to test for control phase change. <0 - Not in use; >0 - Test for control phase changes; =0 - Do not test for control phase changes;
MEVENT	I	C	0 - Do not test for events; 1 - Test for events.

Variable	Input/ Output	Argument/ Common	Definition
NPHASE	I	C	0 - Primary Body Changes. 1 - No Primary Body Changes.
IPFLAG	I/O	C	1 - No Thrust Phase Change has occurred. 2 - Thrust Phase Change has occurred.
JPFLAG	I/O	C	1 - No Primary Body Phase Change has occurred. 2 - Primary Body Phase Change has occurred.
IRECT	O	C	0 - Rectification due to primary body or control phase change. 1 - Trajectory rectification.
ISTEP	O	C	Number of integration step.
NB	I	C	Array containing the bodies to be cosidered in the integration.
NBØD	I	C	Total number of non-zero entries in the NB array.
NPRI	O	C	Number of the primary body.
IPRI	O	C	Location of NPRI in the NB array.
NTPHAS	I/O	C	Number of the current control phase.
NEP	I	C	Number of the ephemeris body.
IEP	O	C	Location of NEP in the NB array.
DRMAX	I	C	Maximum deviation from the reference conic before rectification.
STATEO	I	C	Initial state vector.

Variable	Input/ Output	Argument/ Common	Definition
UTRUE	I/O	C	Position vector relative to the primary body.
VTRUE	I/O	C	Velocity vector relative to the primary body.
ACC	I	C	Trajectory Accuracy level.
FRCA	I	C	Percentage of the semi-major axis of target body to begin closest approach detection.
SCMVAR	I	C	Initial mass variation.
SCMASS	I	C	Initial S/C mass.
THRUST (2, NTPHAS)	I	C	End of the current control phase.
VTRUEM	I	C	Magnitude of VTRUE.
UTRUEM	I	C	Magnitude of UTRUE.
XPRINT	I	C	Time increment of Print (seconds).
G11	I	C	The gravity gradient.
TDUR	I	C	Trajectory stopping time in seconds.
TEVNT	I	C	Event time in seconds.
TCP	O	C	Total integration time.
TREF	I	C	Initial Trajectory Starting time in seconds.
TSTØP	O	C	Time that a stopping criteria has been reached in days.
NRECT	I	C	Number of Rectifications.
ALPHA	I	C	Inverse of semi-major Axis.
BIG	I	C	10^{20}

Variable	Input/ Output	Argument/ Common	Definition
GTAU1	I	C	Thrust noise correlation times.
GTAU2	I	C	Thrust noise correlation times.
NTP	I	C	Number of the target body.
ITP	I	C	Location of target body in the NB array.
QNØISE	I	C	Process noise matrix.
RSTØP	I	C	The stopping radius relative to the target body.
SPHERE	I	C	Array containing all the sphere's of influence.
TSØI	O	C	Time at the sphere of influence of the target body.
TM	I	C	86400 seconds.
TRCA	Ø	C	Time at the closest approach to the target body.
UREL	I	C	Relative position vectors of the spacecraft.
VREL	I	C	Relative velocity vectors of the spacecraft.
DSC	I/O	C	The blank common array where the following flags (LØCH to LØCX) are used to locate data.
LØCH	I	C	Integration step-size (h).
LØCM	I	C	Spacecraft mass (*m).
LØCFI	I	C	F matrix (F).
LØCPR	I	C	Trajectory integration print time (t_{PR}).

Variable	Input/ Output	Argument/ Common	Definition
LØCPT	I	C	Trajectory print time (t_p).
LØCDM	I	C	Mass variation (δm).
LØCDT	I	C	Differential equations for events and print ($*F_i$).
LØCDY	I	C	Differential equations for the reference (F_i).
LØCET	I	C	Event integration time (t_e).
LØCFØ	I	C	Location of the input covariance.
LØCR	I	C	Location of the stored position magnitudes (r_i).
LØCT	I	C	Location of the stored position trajectory times (t_i).
LØCTC	I	C	Location of the output transition matrix or covariance ($*P$ or $*\Phi$).
LOCYC	I	C	Integrated equations for the reference (Y_{k+1}).
LØCYP	I	C	Integrated equations working array (Y_k).
LØCYT	I	C	Integrated equations for events and print ($*Y_{k+1}$).
LØCX	I	C	Trajectory time (X_k).
MEQ	I	C	Total number of equations to be integrated.
MEQ8	I	C	MEQ-8.
MEQS	I	C	$\sqrt{\text{MEQ8}}$

Variable	Input/ Output	Argument/ Common	Definition
KUTØFF	Ø	C	This flag indicates to the mode calling TRAJ why the trajectory was terminated. Other than termination on final time and an event, the other terminations, closest approach, sphere of influence and stopping radius are not always satisfied. Therefore this multi-valued flag gives a different value for the actual stopping condition. The following table shows the relationship between KUTØFF and ISTØP.

Requested	ISTØP	Actual	KUTØFF
Final Time	1	Final Time	1
Closest Approach	2	Final Time	2
Sphere of Influence	3	Final Time	3
Stopping Radius	4	Final Time	4
Closest Approach	2	Closest Approach	5
Sphere of Influence	3	Closest Approach	6
Sphere of Influence	3	Sphere of Influence	7
Stopping Radius	4	Stopping Radius	8
Event Time	NA	Event Time	9

Local Variables:

Variable	Definition
HEVNT	Event integration step-size.
HPRNT	Print integration step-size.
IRSTP	Indicates termination for determining KUTØFF.

The following variables are used in assigned GØ TØ statements and are in the TRAJ1 common block. When these statements are used in FLIGHT, there are implicit tests made. The majority of the tests are made in PATH. ITRAJ, IPHAS0, IPHAS1, IPHAS2, JPHAS1, JPHAS2, JPHAS3, JTEST, KSTØP, LØCAL, MSTØP, NSTØP, IEVNT1, IEVNT2, IEVNT3, INTEG2, INTEG3, IPHASE, IPRT, IEVENT.

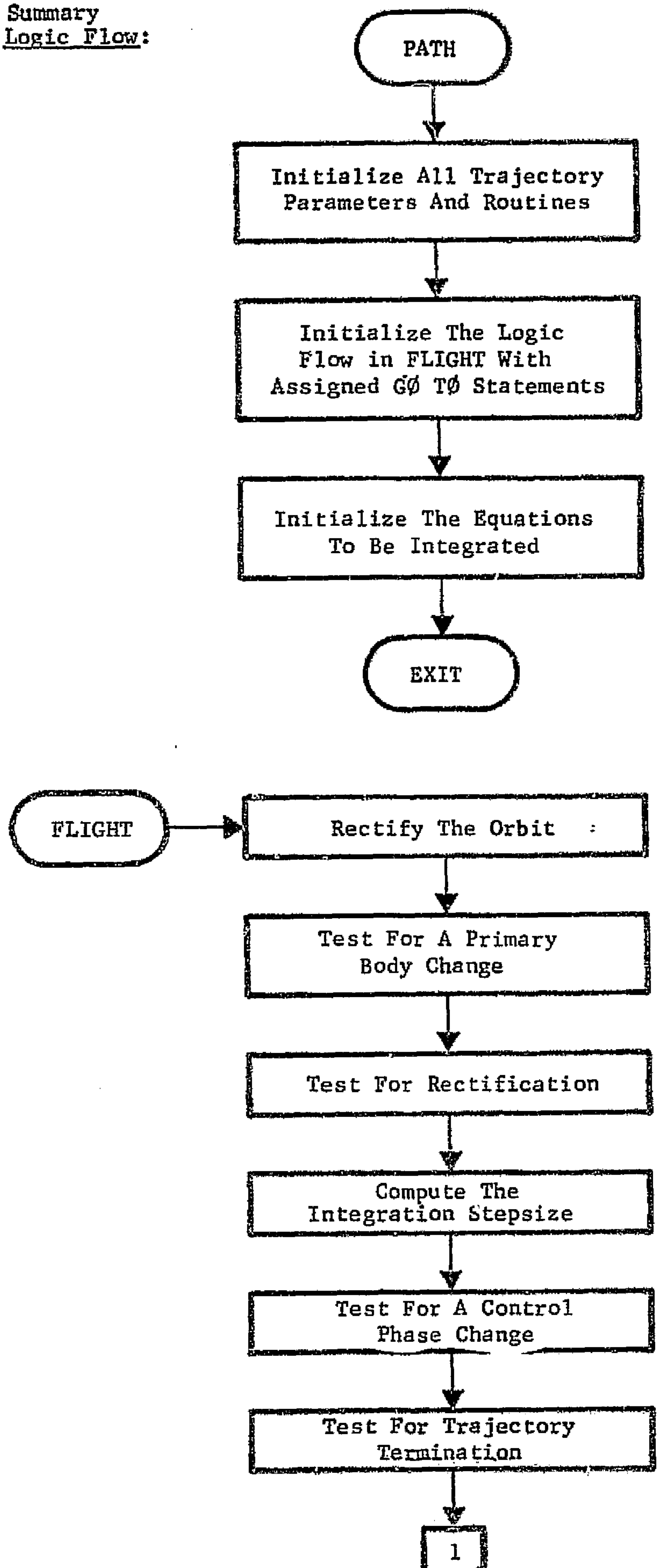
<u>Subroutines Called</u>: CØPY, DPHI, FIND, FIND1, FIND3, IDENT, LØCATE, MØTIØN, NEWTØN, PDØT, PRINTT, RUNG2, RUNG4, SETUP, UDØTV, VECMAG, ZERØM

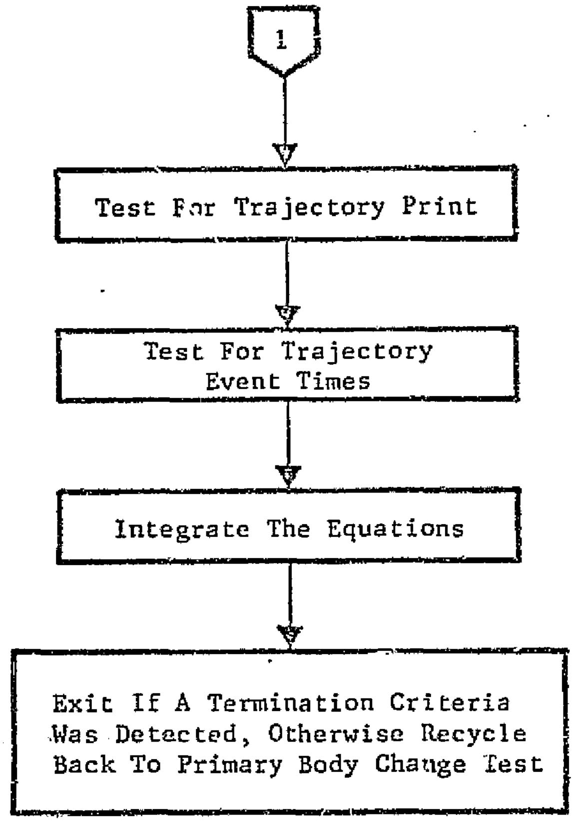
<u>Calling Subroutines</u>: TRAJ

<u>Common Blocks</u>: (BLANK), CØNST, EPHEM, TIME, TRAJ1, TRAJ2, WØRK

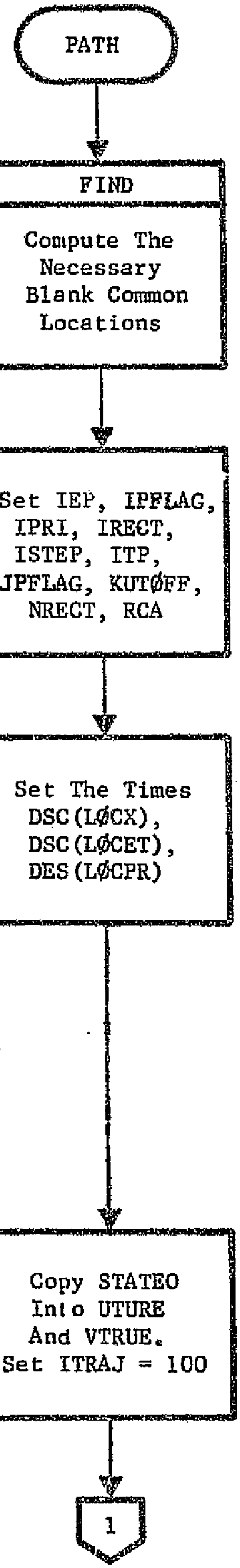
<u>Logic Flow</u>: The functional flow of PATH and FLIGHT is given on the next two pages, followed by a more detailed logic flow.

Summary
Logic Flow:

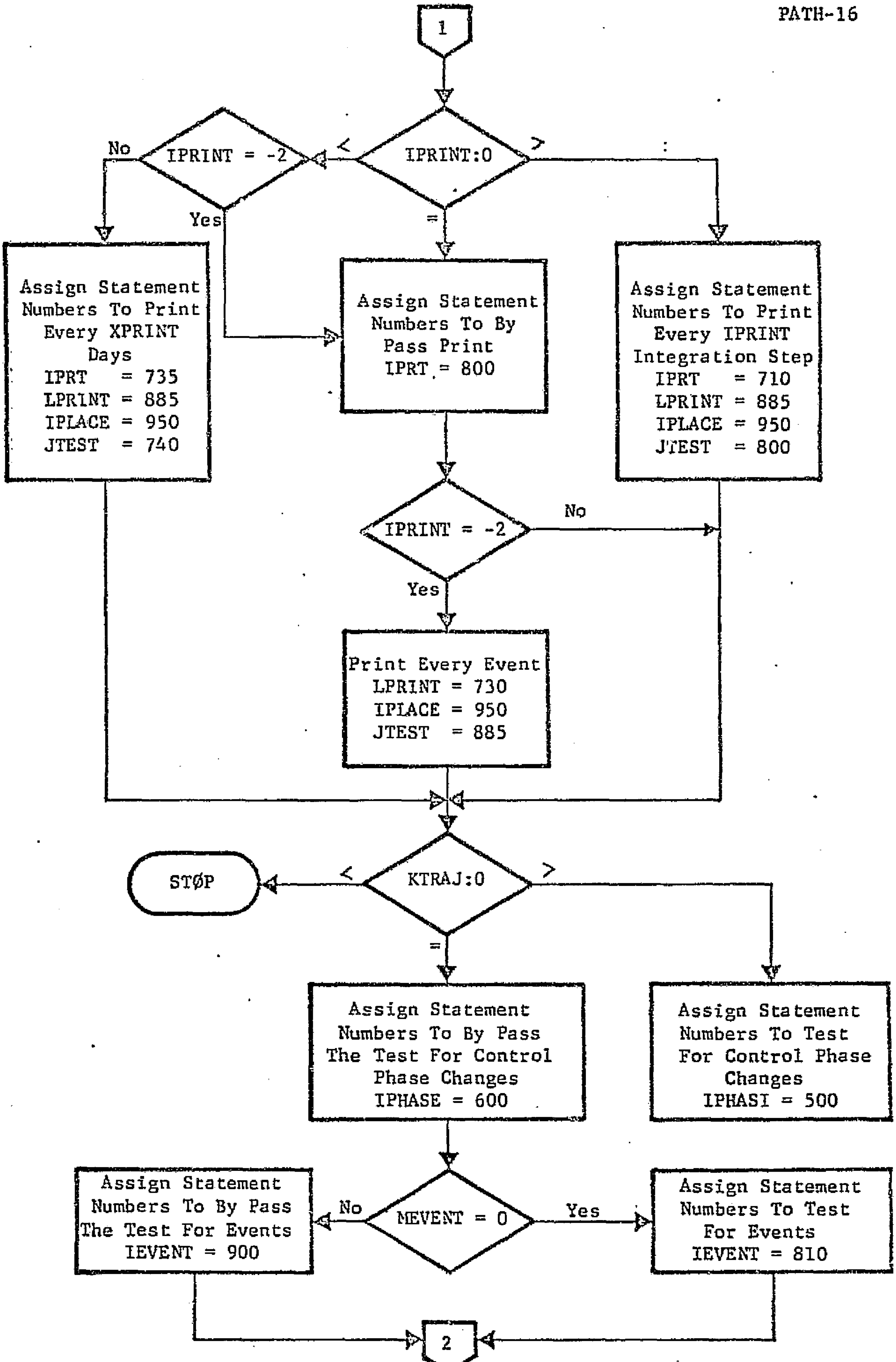


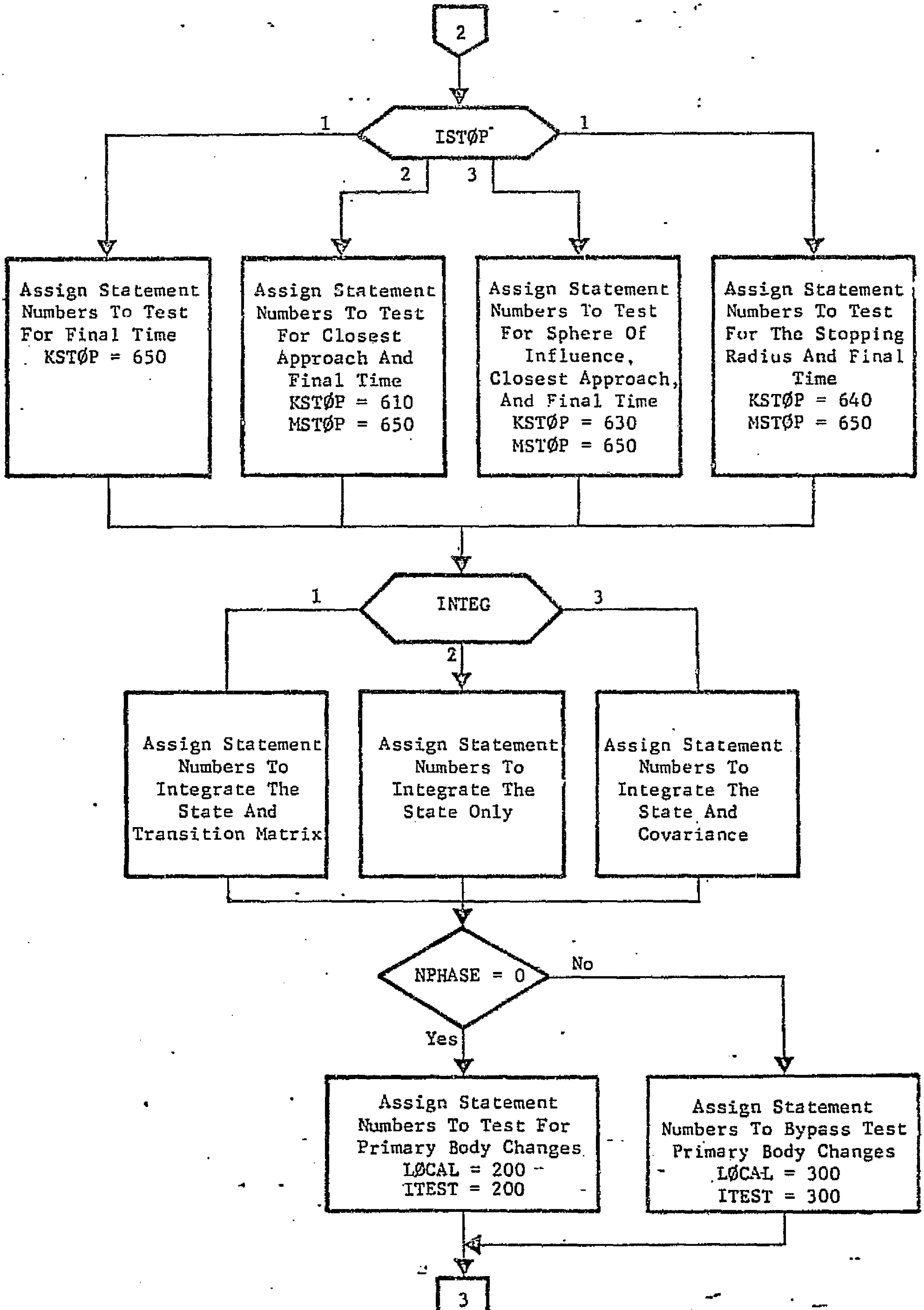

1
Test For Trajectory Print
Test For Trajectory Event Times
Integrate The Equations
Exit If A Termination Criteria Was Detected, Otherwise Recycle Back To Primary Body Change Test

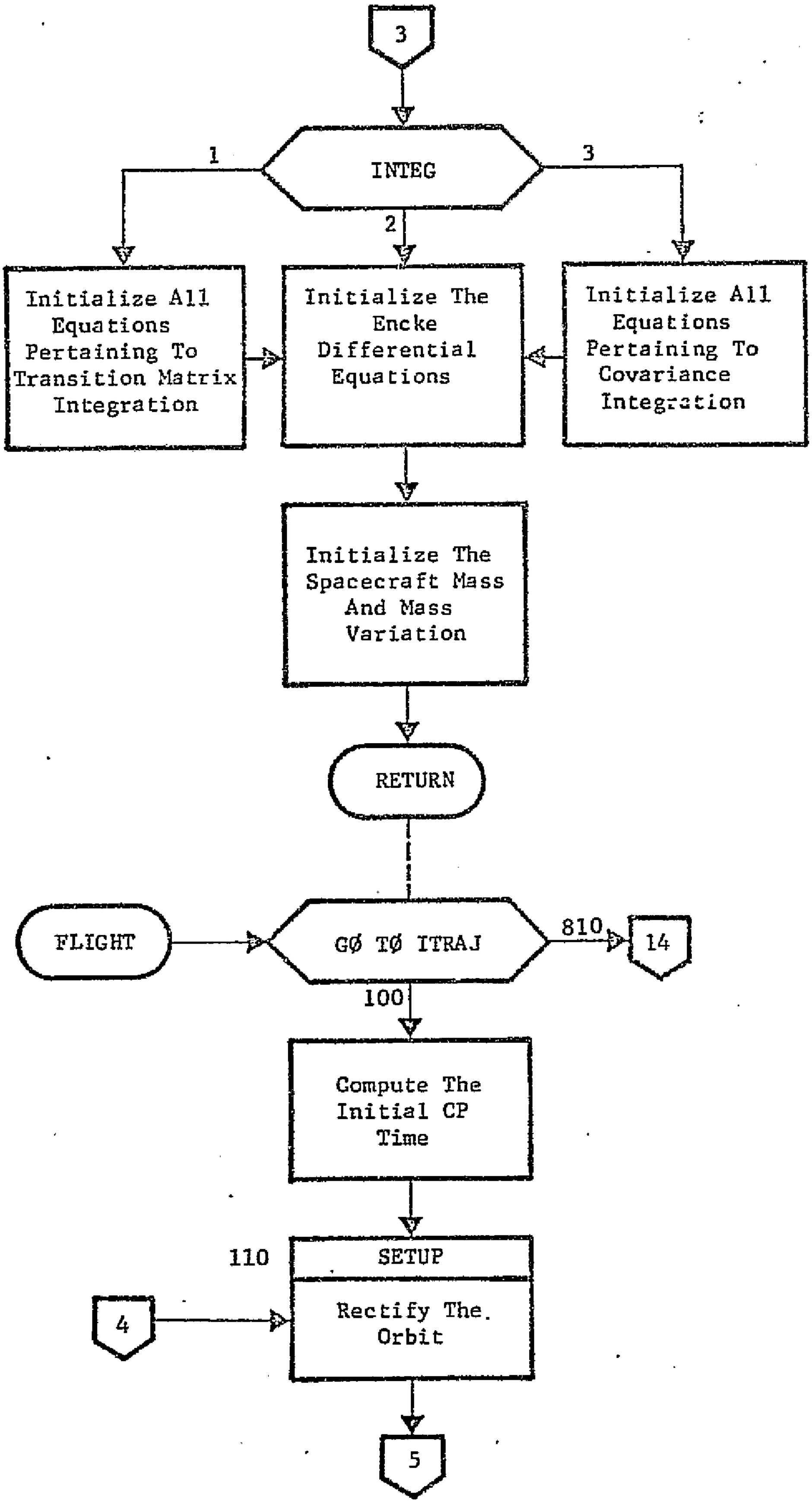
Detailed
Logic Flow:

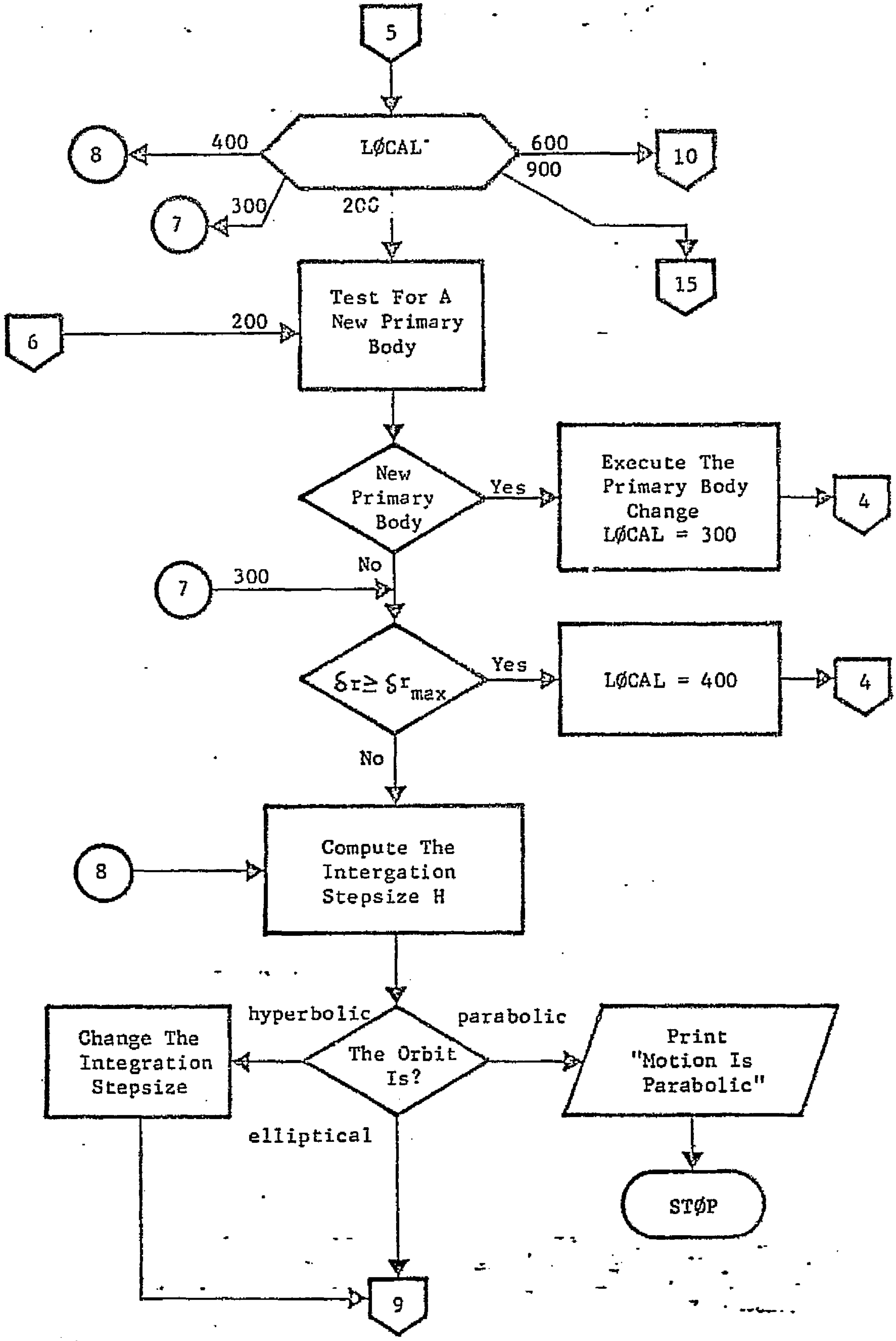


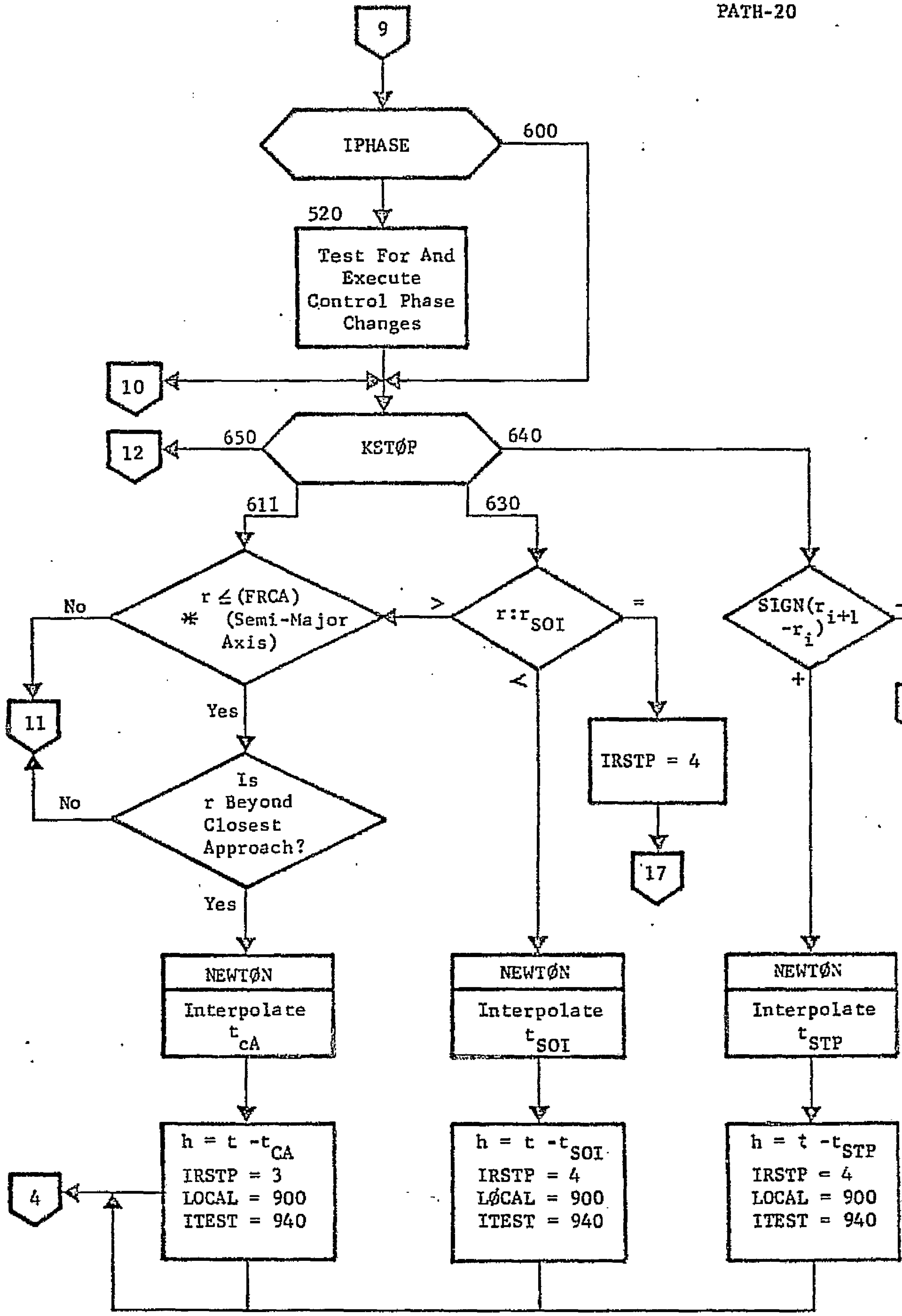
PATH-16

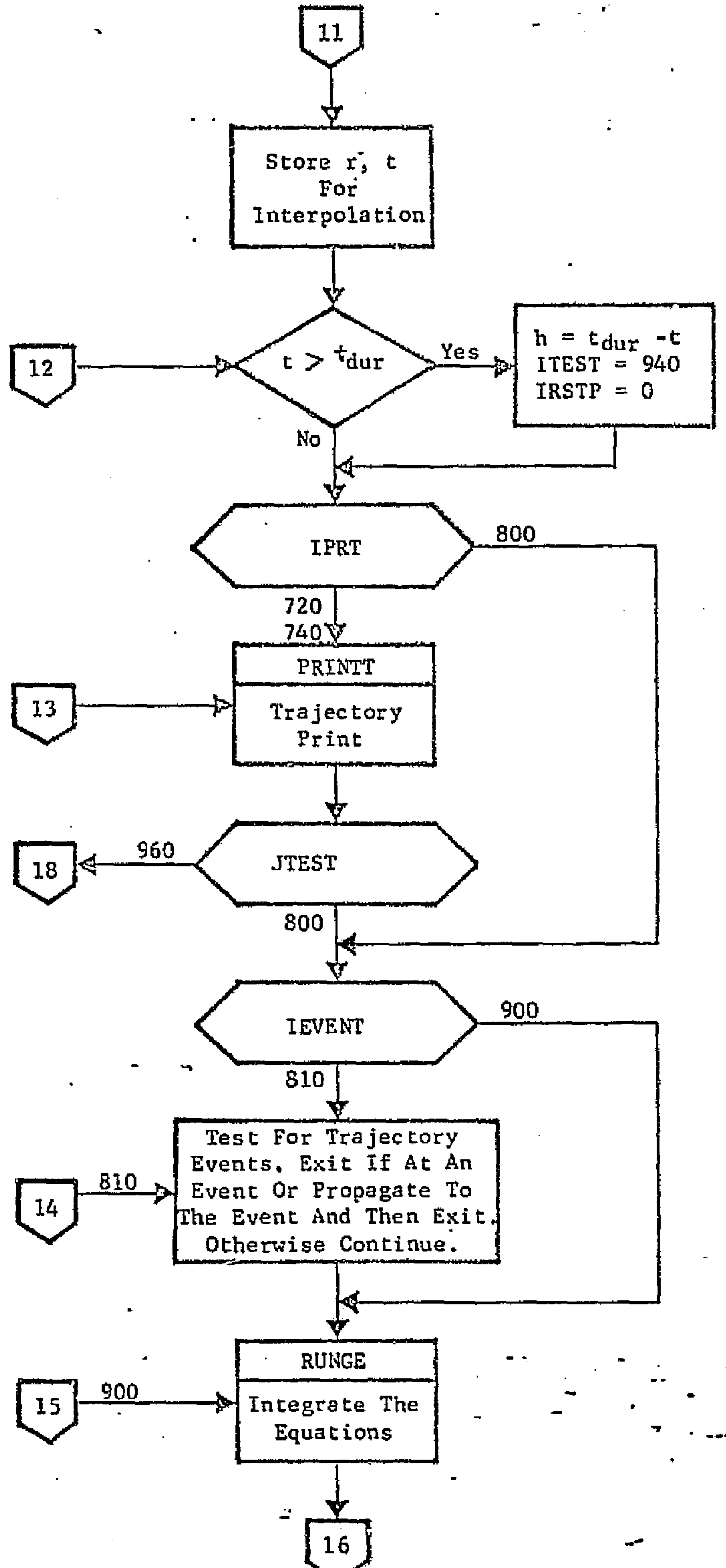

1
IPRINT:0
<
>
=
IPRINT = -2
No
Yes
Assign Statement Numbers To Print Every XPRINT Days
IPRT = 735
LPRINT = 885
IPLACE = 950
JTEST = 740
Assign Statement Numbers To By Pass Print
IPRT = 800
Assign Statement Numbers To Print Every IPRINT Integration Step
IPRT = 710
LPRINT = 885
IPLACE = 950
JTEST = 800
IPRINT = -2
No
Yes
Print Every Event
LPRINT = 730
IPLACE = 950
JTEST = 885
KTRAJ:0
<
>
=
STØP
Assign Statement Numbers To By Pass The Test For Control Phase Changes
IPHASE = 600
Assign Statement Numbers To Test For Control Phase Changes
IPHASI = 500
MEVENT = 0
No
Yes
Assign Statement Numbers To By Pass The Test For Events
IEVENT = 900
Assign Statement Numbers To Test For Events
IEVENT = 810
2

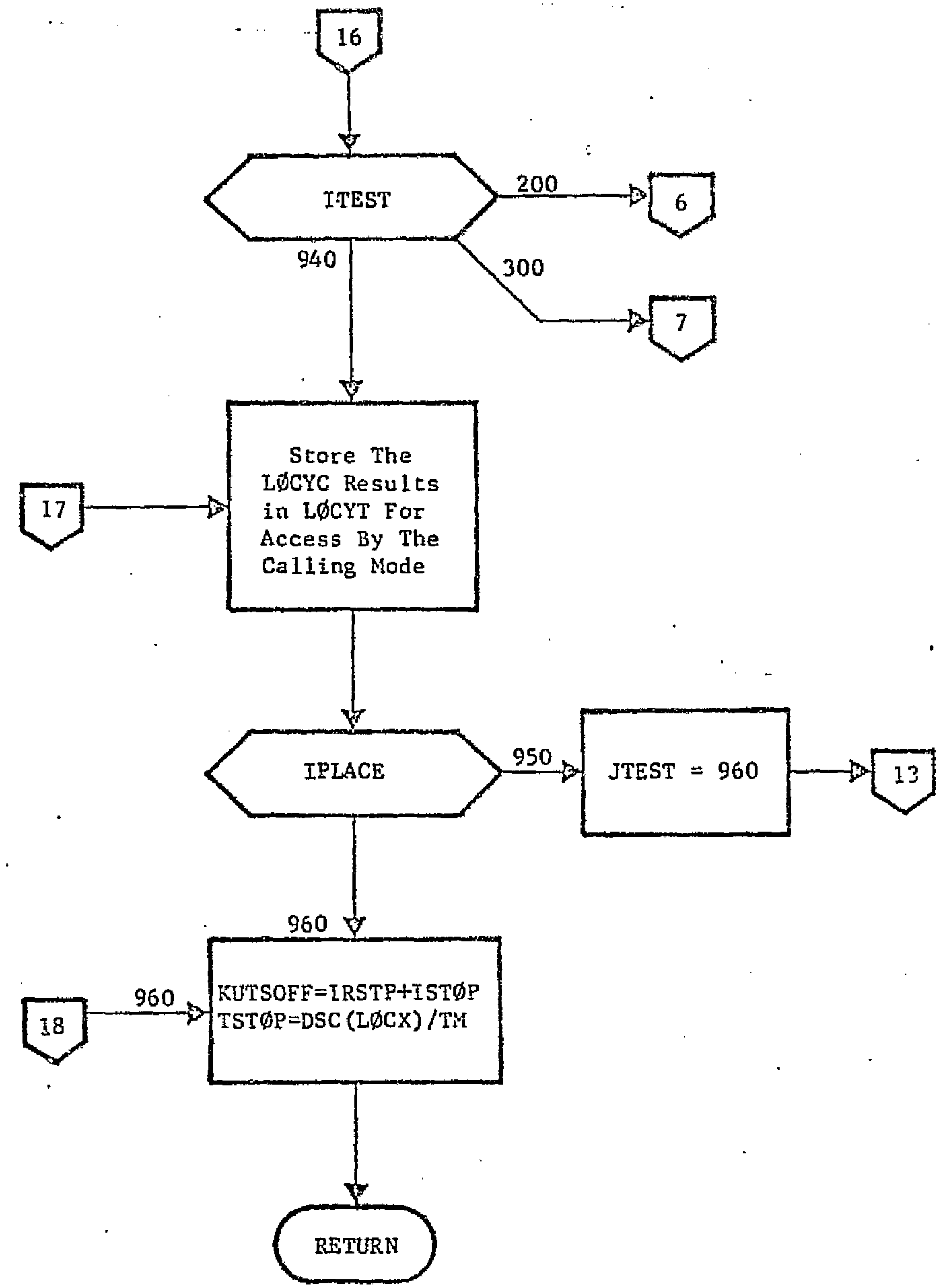

2
ISTØP
1
1
2
3
Assign Statement Numbers To Test For Final Time KSTØP = 650
Assign Statement Numbers To Test For Closest Approach And Final Time KSTØP = 610 MSTØP = 650
Assign Statement Numbers To Test For Sphere Of Influence, Closest Approach, And Final Time KSTØP = 630 MSTØP = 650
Assign Statement Numbers To Test For The Stopping Radius And Final Time KSTØP = 640 MSTØP = 650
INTEG
1
2
3
Assign Statement Numbers To Integrate The State And Transition Matrix
Assign Statement Numbers To Integrate The State Only
Assign Statement Numbers To Integrate The State And Covariance
NPHASE = 0
No
Yes
Assign Statement Numbers To Test For Primary Body Changes LØCAL = 200 ITEST = 200
Assign Statement Numbers To Bypass Test Primary Body Changes LØCAL = 300 ITEST = 300
3


3
INTEG
1
2
3
Initialize All Equations Pertaining To Transition Matrix Integration
Initialize The Encke Differential Equations
Initialize All Equations Pertaining To Covariance Integration
Initialize The Spacecraft Mass And Mass Variation
RETURN
FLIGHT
GØ TØ ITRAJ
810
14
100
Compute The Initial CP Time
110
SETUP
Rectify The Orbit
4
5


5
LØCAL
400
8
600
10
900
300
7
200
15
Test For A New Primary Body
6
200
New Primary Body
Yes
Execute The Primary Body Change LØCAL = 300
4
No
7
300
$\delta r \geq \delta r_{max}$
Yes
LØCAL = 400
4
No
Compute The Intergation Stepsize H
8
hyperbolic
parabolic
Change The Integration Stepsize
The Orbit Is?
Print "Motion Is Parabolic"
elliptical
STØP
9


9
IPHASE
600
520
Test For And Execute Control Phase Changes
10
KSTØP
650
640
12
611
630
r ≤ (FRCA) * (Semi-Major Axis)
No
>
r:r SOI
=
SIGN(r i+1 -r i)
-
+
<
11
Yes
IRSTP = 4
11
Is r Beyond Closest Approach?
No
17
Yes
NEWTØN
Interpolate t cA
NEWTØN
Interpolate t SOI
NEWTØN
Interpolate t STP
h = t -t CA
IRSTP = 3
LOCAL = 900
ITEST = 940
4
h = t -t SOI
IRSTP = 4
LØCAL = 900
ITEST = 940
h = t -t STP
IRSTP = 4
LOCAL = 900
ITEST = 940


11
Store r, t
For
Interpolation
12
t > t_{dur}
Yes
h = t_{dur} -t
ITEST = 940
IRSTP = 0
No
IPRT
800
720
740
PRINTT
13
Trajectory
Print
18
960
JTEST
800
IEVENT
900
810
14
810
Test For Trajectory
Events. Exit If At An
Event Or Propagate To
The Event And Then Exit.
Otherwise Continue.
RUNGE
15
900
Integrate The
Equations
16


16
ITEST
200
6
940
300
7
Store The LØCYC Results in LØCYT For Access By The Calling Mode
17
IPLACE
950
JTEST = 960
13
960
KUTSOFF=IRSTP+ISTØP
TSTØP=DSC(LØCX)/TM
18
960
RETURN

3.5.13 Subroutine: PDØT (T, DS, DP, M, N, LØC)

Purpose: To compute the time derivative of the state covariance (P)

Method: $\dot{P} = FP + PF^T + Q$

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
T	I	A	Trajectory time
DS	I	A	Independent variables
DP	Ø	A	Differential equations
M	I	A	Number of rows in DS and DP
N	I	A	Number of columns in DS and DP
LØC	I	A	Routing flag
INTEG	I	C	Set = 1 Propagate the state and Transition Matrix Set = 2 Propagate the state Set = 3 Propagate the state and state covariance
IAUGDC	I	C	Flag indicating the augmentation of the STM and covariance matrix
IRECT	I	C	Index used to check whether the current call to PDØT is for rectification purposes only (i.e. IRECT = 1)

Local Variables

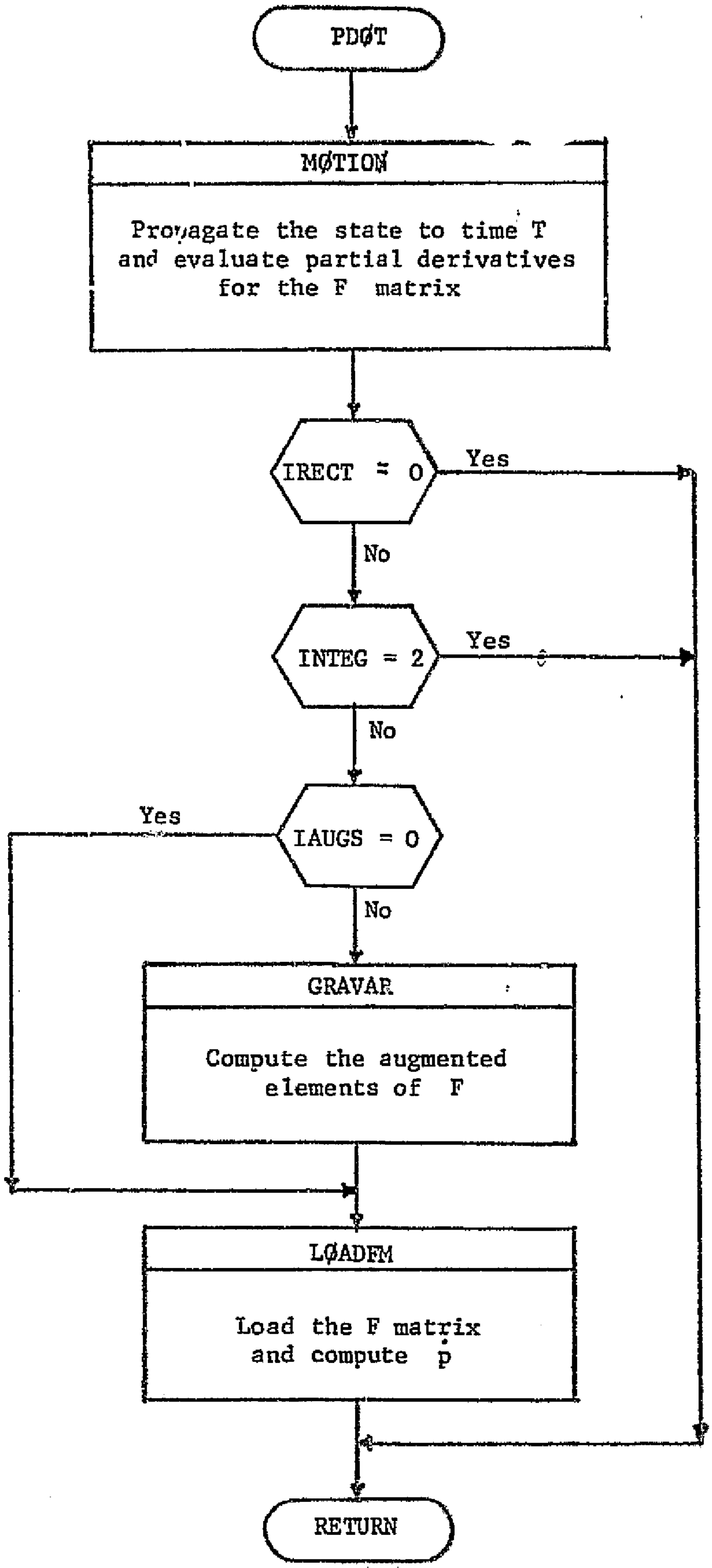
Variable	Definition
IAUGS	Index used to check whether the F matrix needs to be augmented

Calling Subroutines: NUMIN

Subroutines Called: MØTIØN, LØADFM, GRAVAR

Common Blocks: TRAJ2

Logic Flow:


PDØT
MØTION
Propagate the state to time T and evaluate partial derivatives for the F matrix
IRECT = 0
Yes
No
INTEG = 2
Yes
No
IAUGS = 0
Yes
No
GRAVAR
Compute the augmented elements of F
LØADFM
Load the F matrix and compute $\dot{p}$
RETURN

3.5.14 Function: PØWER (R, TT)

Purpose: PØWER computes the power available to the thrusters of the low thrust spacecraft for solar electric and nuclear propulsion.

Method: The power is computed from the following expression.

$$P = \begin{cases} P_o \left[\dfrac{A_1}{r^2} + \dfrac{A_2}{r^{5/2}} + \dfrac{A_3}{r^3} + \dfrac{A_5}{r^5} \right] * \exp\left[-P_L(t-t_{DL})\right] - P_{HK}, & \text{solar electric} \\ P_{max}, \text{ if } P > P_{max} \text{ or } r < r_{min}, & \text{solar electric} \\ P_o \exp\left[-P_L(t-t_{DL})\right] - P_{HK}, & \text{nuclear} \end{cases}$$

P_o - Power available (at 1 AU for solar, at energization for nuclear)

A_i - (Empirical) Constants defining solar array characteristics

r - Heliocentric position magnitude of the S/C

P_L - Power decay constant

t - Time from epoch

t_{DL} - Time delay

P_{HK} - Housekeeping power

P_{max} - Maximum allowable solar electric power

r_{min} - Heliocentric distance for which P is less than P_{max}

Input/Output:

Variable	Input Output	Argument/ Common	Definition
R	I	A	Heliocentric distance in A.U. (r)
TT	I	A	Trajectory time in seconds (t)
PØWERO	I	C	P_o (Equivalenced to ENGINE(1))
PHK	I	C	P_{HK} (Equivalenced to ENGINE(2))
PMAX	I	C	P_{max} (Equivalenced to ENGINE(3))
A1	I	C	A_1 (Equivalenced to ENGINE(4))
A2	I	C	A_2 (Equivalenced to ENGINE(5))
A3	I	C	A_3 (Equivalenced to ENGINE(6))
A4	I	C	A_4 (Equivalenced to ENGINE(7))
A5	I	C	A_5 (Equivalenced to ENGINE(8))
RMIN	I	C	r_{min} (Equivalenced to ENGINE(9))
PLØSS	I	C	P_L (Equivalenced to ENGINE(12))
TDL	I	C	t_{DL} (Equivalenced to ENGINE(13))

Variable	Input/ Output	Argument/ Common	Definition
IENRGY	I	C	Flag that determines the type of power 0 - nuclear power 1 - solar electric power
PØWER	Ø	F*	Power available to the thrusters.

Local Variables:

Variable	Definition
SR	$\frac{1}{\sqrt{r}}$
SCALE	$\frac{A_1}{r^2} + \frac{A_2}{r^{5/2}} + \frac{A_3}{r^3} + \frac{A_4}{r^{7/2}} + \frac{A_5}{r^5}$

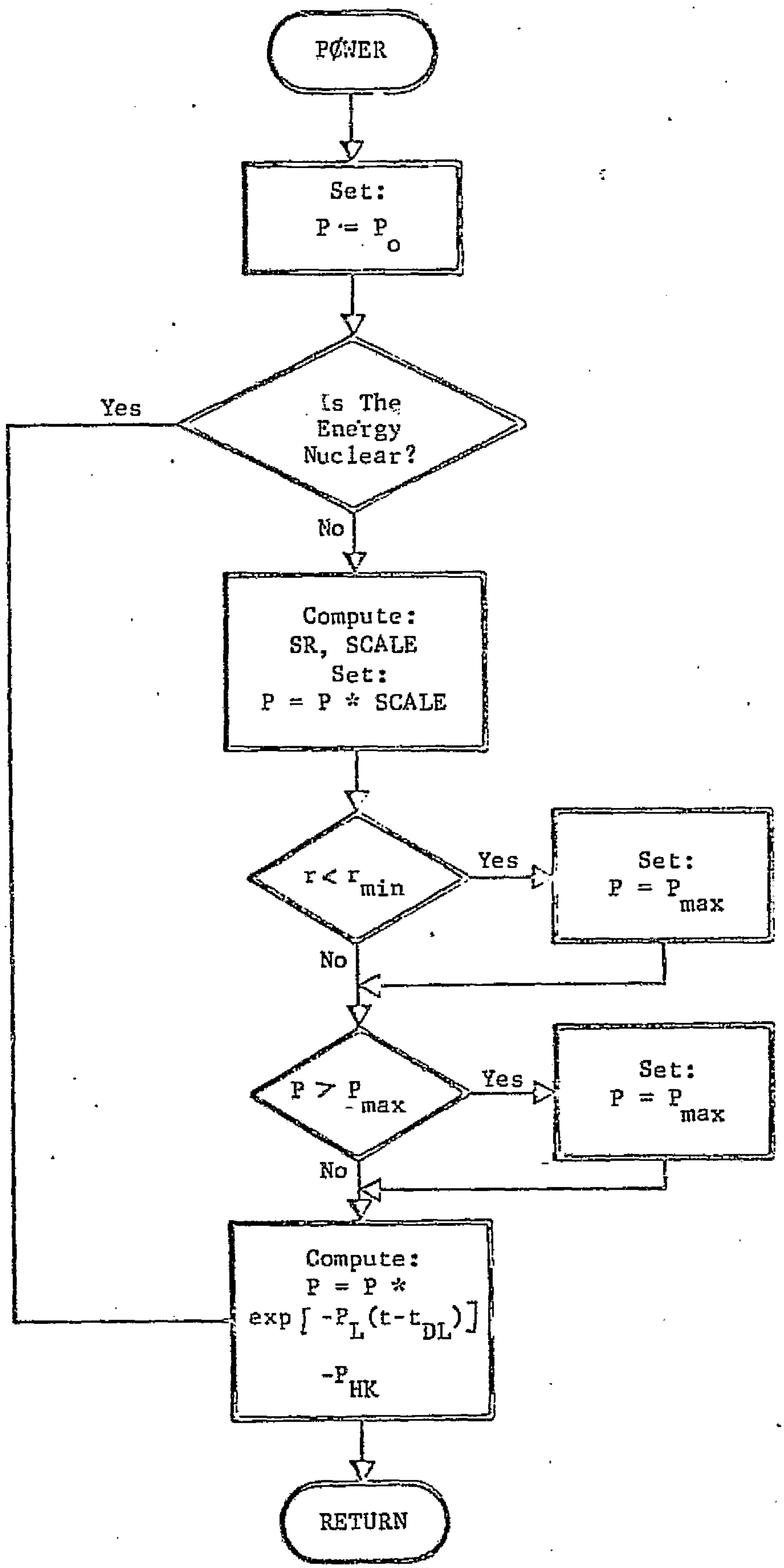
Subroutines Called: None

Calling Subroutines: EP

Common Blocks: CØNST, TRAJ1, TRAJ2

*Function Value Output

Logic Flow:


PØWER
Set:
P = P_o
Is The Energy Nuclear?
Yes
No
Compute:
SR, SCALE
Set:
P = P * SCALE
r < r_min
Yes
Set:
P = P_max
No
P > P_max
Yes
Set:
P = P_max
No
Compute:
P = P *
exp [-P_L (t-t_DL)]
-P_HK
RETURN

3.5.15 Subroutine: PRINTT (TT, MASS)

Purpose: To print trajectory and spacecraft related information.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
NTPHAS	I	C	Number of the current thrust phase.
NPRI	I	C	Number of the current primary body.
NEP	I	C	Number of the ephemeris body.
NTP	I	C	Number of the target body.
PLANET	I	C	Array containing the names of the planets.
MASS	I	A	Current spacecraft mass.
WPØWER	I	C	Current power available to the spacecraft for thrust.
TT	I	A	Trajectory time in days.
TDUR	I	C	Trajectory termination time in seconds.
EPØCH	I	C	Trajectory initial time (Julian days).
TM	I	C	86400. seconds.
APRIM	I	C	Acceleration vector due to the gravity of the primary body.
THRACC	I	C	Acceleration vector due to thrust.
RPACC	I	C	Acceleration vector due to radiation pressure.

Variable	Input/ Output	Argument/ Common	Definition
IPFLAG	I	C	Flag that indicates control phase change.
JPFLAG	I	C	Array containing only the names of the planets included in the integration.
APERT	I	C	Matrix containing the acceleration vectors due to the gravity of the non-primary bodies.
UREL	I	C	Matrix of spacecraft position vectors relative to the bodies considered in the integration.
URELM	I	C	Array containing magnitudes of the position vectors.
VREL	I	C	Matrix of spacecraft velocity vectors relative to the bodies considered in the integration.
VRELM	I	C	Array containing magnitudes of the velocity vectors.
MPLAN	I	C	Total number of bodies included in the integration.
THRUST	I	C	Array containing the thrust control. To locate information for the current control phase NTPHAS is used as follows: THRUST (i, NTPHAS) where i is the desired information.

<u>Local Variables</u>:

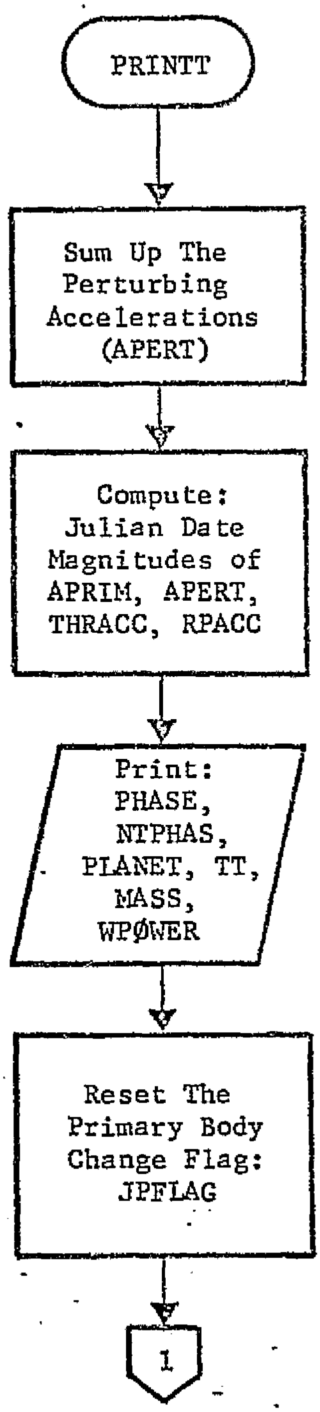
Variable	Definition
WØRK	Temporary storage array.
PHASE	Array that contains headings for control and primary body changes.

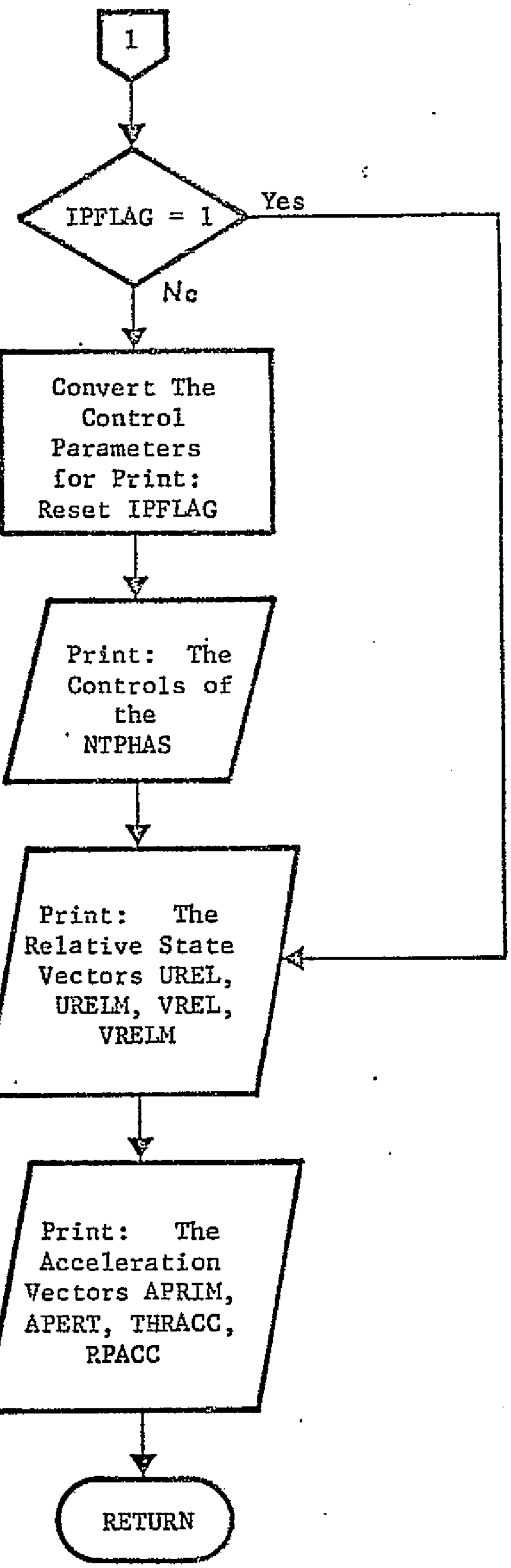
Subroutines Called: None

Calling Subroutines: PATH, MEASPR

Common Blocks: CØNST, EPHEM, TIME, TRAJ1, TRAJ2

Logic Flow:




1
IPFLAG = 1
Yes
No
Convert The Control Parameters for Print: Reset IPFLAG
Print: The Controls of the NTPHAS
Print: The Relative State Vectors UREL, URELM, VREL, VRELM
Print: The Acceleration Vectors APRIM, APERT, THRACC, RPACC
RETURN

3.5.16 Subroutine: RPRESS (CMASS)

Purpose: RPRESS computes the effective acceleration acting on a spacecraft due to radiation pressure.

Method: The effective acceleration is computed from the following expression.

$$\underline{a}_R = \frac{(1.024 \times 10^8)\ C_r A}{m\ r^2} \cdot \frac{\underline{r}}{r}$$

$\underline{r}$ - heliocentric position vector of the spacecraft.

m - spacecraft mass.

$C_r A$ - coefficient of reflectivity multiplied by the effective area of the solar array.

In the event that $r \leq r_{min}$, where r_{min} is the distance at which the solar electric power is a maximum, the effective cross sectional area of the solar array is changed by tilting (or folding) them. Therefore, the effective acceleration is reduced,

$$\underline{a}_R = \underline{a}_R \cdot \cos\alpha$$

where α is the off-sun tilt angle.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
CMASS	I	A	Current spacecraft mass.

Variable	Input/ Output	Argument/ Common	Definition
CRA	I	C	$C_r A$ (Equivalenced to ENGINE(15)).
CTILT	I	C	$\cos \alpha$ (Equivalenced to ENGINE(16)).
RMIN	I	C	r_{min} (Equivalenced to ENGINE(9)).
URELM(1)	I	C	Heliocentric position of the spacecraft.
UREL(I, 1)	I	C	Heliocentric position vector of the spacecraft.
RPACC	Ø	C	$\underline{a}_r$

Local Variables:

Variable	Definition
RPA	$\lvert \underline{a}_r \rvert$

Subroutines Called: None

Calling Subroutine: MØTIØN

Common Blocks: CØNST, TRAJ1

3.5.17 Subroutine: SØLAR (JDATE)

Purpose: To compute the position and velocity of the planets.

Method: None

Input/Output:

Variables	Input Output	Argument/ Common	Definition
NB	I	C	Array of bodies for which the position and velocity are to be computed.
JDATE	I	A	Julian Date at which the position and velocity are to be computed
UP	Ø	C	Array of position vectors
VP	Ø	C	Array of velocity vectors

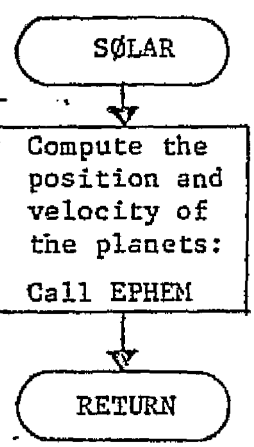
Local Variables: None

Subroutines Called: EPHEM

Calling Subroutine: MØTIØN

Common Blocks: TRAJ1, TRAJ2

Logic Flow:



3.6 Utility Routines

A number of subroutines and function routines are used in each mode that are (1) standard to many scientific computer programs, or (2) common to more than one MAPSEP mode. These utility routines are described in this Section. The first group (3.6.1) contain relatively minor and straightforward routines that perform matrix manipulation and vector operations. The second group (3.6.2 through 3.6.11) describe more complex utility routines, all of which apply standard mathematical techniques to compute specific parameters required by MAPSEP.

3.6.1 Minor Subroutines

The following utility routines are straightforward in usage and internal computation. Their description consists of name (and any entry points), input and output arguments, and function. No common blocks are contained in these routines and all are subroutines except UDØTV and VECMAG which are function routines.

Subroutine (Entry Points)	Arguments	Function
ADD	A, B, C, M, N	ADD performs the matrix operation $[C]_{MxN} = [A]_{MxN} + [B]_{MxN}$ matrices.
CØPY(ICØPY)	A, B, M, N	CØPY copies a real matrix A into matrix B, where A and B are MxN. ICØPY assumes A and B are integer matrices.
CØPYT	CT, C, M, N	Copies the transpose of the matrix CT into matrix C, where CT is NxM and C is MxN.
EIGENV	A, N, FØD, W2, V	EIGENV computes the eigenvalues and eigenvectors of a N X N matrix A, using Jocobi's method of successive rotations. FØD is the tolerance for the off diagonal elements of A. The eigenvalues and eigenvectors are returned in the vector arrays W2 and V, respectively.
IDENT	C, N	Creates an NxN identity matrix C.
INVSQM	A, N, XB, RTEST, IX, IY	INVSQM inverts an NxN matrix A by the Gauss-Jordan elimination method. The results are returned in A. INVSQM requires four Nx1 vectors, XB, RTEST, IX and IY, for temporary storage (to keep core requirements to a minimum).
JØBTLE	None	JØBTLE is used by GØDSEP to eject a page and to print out the job title, a row of asterisks and the trajectory time.
MATØUT	A, NRØW, NCØL, LABEL	MATØUT prints a matrix A, NRØWxNCØL, with a 6 character Hollerith label, LABEL.
MMAB (AMAB)	A, B, C, M, L, N	MMAB performs the matrix operation $[C]_{MxN} = [A]_{MxL} * [B]_{LxN}$. AMAB performs the matrix operation $[C]_{MxN} = [C]_{MxN} + [A]_{MxL} * [B]_{LxN}$

Subroutine (Entry Points)	Arguments	Function
MMABAT (AMABAT)	A, B, C, M, L, N	MMABAT performs the matrix operation $[C]_{MxM} = [A]_{MxL} * [B]_{LxL} * [A]^T_{MxL}$ (Note: N is not used). AMABAT performs the matrix operation $[C]_{MxM} = [C]_{MxM} + [A]_{MxL} * [B]_{LxL} * [A]^T_{MxL}$.
MMABT (AMABT)	A, B, C, M, L, N	MMABT performs the matrix operation $[C]_{MxN} = [A]_{MxL} * [B]^T_{NxL}$. AMABT performs the matrix operation $[C]_{MxN} = [C]_{MxN} + [A]_{MxL} * [B]^T_{NxL}$.
MMATB (AMATB)	A, B, C, M, L, N	MMATB performs the matrix operation $[C]_{MxN} = [A]^T_{LxM} * [B]_{LxN}$. AMATB performs the matrix operation $[C]_{MxN} = [C]_{MxN} + [A]^T_{LxM} * [B]_{LxN}$.
MMATBA (AMATBA)	A, B, C, M, L, N	MMATBA performs the matrix operation $[C]_{MxM} = [A]^T_{LxM} * [B]_{LxL} * [A]_{LxM}$. Note: N is not used. AMATBA performs the matrix operation $[C]_{MxM} = [C]_{MxM} + [A]^T_{LxM} * [B]_{LxL} * [A]_{LxM}$.
MMATBT (AMATBT)	A, B, C, M, L, N	MMATBT performs the matrix operation $[C]_{MxN} = [A]^T_{LxM} * [B]^T_{NxL}$. AMATBT performs the matrix operation $[C]_{MxN} = [C]_{MxN} + [A]^T_{LxM} * [B]^T_{NxL}$.

Subroutine (Entry Points)	Arguments	Function
NEGMAT	A, C, M, N	NEGMAT negates a matrix such that $[C]_{MxN} = - [A]_{MxN}$.
SCALE	FACTØR, A, M, N, B	SCALE multiplies a matrix A by a scalar FACTØR and returns the result in a matrix B, $[B]_{MxN}$ = FACTØR * $[A]_{MxN}$.
SDVAR (VARSD)	CØVIN, CØVØUT, N	SDVAR takes an NxN matrix CØVIN of standard deviations and correlation coefficients, and operates on the lower triangle of CØVIN to create a full covariance matrix CØVØUT. VARSD takes an NxN covariance matrix CØVIN and operates on the upper triangle to create a matrix CØVØUT, where only the upper triangle contains the correlation coefficients, the diagnal the standard deviation and the lower triangle remains unchanged.
SUB	A, B, C, M, N	SUB subtracts matrix B from matrix A and returns the results as matrix C. The dimensions of A, B, and C are MxN.
SUBT	A, B, C, M, N	SUBT subtracts matrix B^T from matrix A and returns the results as matrix C. The dimensions of A and C are MxN, B is NxM.

Subroutine (Entry Points)	Arguments	Function
SYMTRZ (SYMLØ, SYMUP)	PSYM, N	SYMTRZ takes an NxN matrix PSYM and makes it symmetric by averaging each corresponding off-diagonal pair. SYMLØ takes an NxN matrix PSYM and makes the upper triangle equal to the lower triangle. SYMUP takes an NxN matrix and makes the lower triangle equal to the upper triangle.
UDØTV	U, V	UDØTV performs the vector operation $\underline{U} \cdot \underline{V}$, for three dimensional vectors.
UNITV	U, UV	UNITV take a three dimensional vector U and makes it a unit vector UV.
UXV	U, V, W	UXV performs the vector operation $\underline{UxV} = \underline{W}$, for three dimensional vectors.
VECMAG	U	VECMAG computes the magnitude of a three dimensional vector.
ZERØM	A, MRØW, MCØL	ZERØM creates a MRØW x MCØL null matrix A.

3.6.2 Subroutine: BPLANE (R, V, TO, NTF)

Purpose: To compute osculating B-plane parameters (Section 4.2 of Reference 1) relative to a specified body from a cartesian state vector. Also, BPLANE computes a number of other osculating parameters, e.g., radius of closest approach (r_{ca}).

Method: Given the spacecraft planeocentric ecliptic position and velocity vectors, $\underline{r}$ and $\underline{v}$ respectively, at time t relative to a target body, compute the B-plane parameters, $\underline{B} \cdot \underline{T}$, $\underline{B} \cdot \underline{R}$, and the associated conic elements. In order that all B-plane parameters are computed, the target relative osculating conic should be a hyperbola with its radius of closest (r_{ca}) approach inside the sphere of influence (SOI) of the target body. Assuming that closest approach is within the sphere of influence (r_{SOI}). Then using the orbital elements (a, e, i, Ω, ω, M), calculated from the conic formulas of Section 3.6.4

$$r_{CA} = a(1 - e)$$

$$V_{CA} = \sqrt{\mu\left(\frac{2}{r_{CA}} - \frac{1}{a}\right)}$$

$$V_{hp} = \sqrt{\frac{\mu}{|a|}}$$

$$p = a(1 - e^2)$$

where V_{hp} is the hyperbolic excess velocity. The R,S,T coordinate system for the B-plane parameters is defined as

$$\hat{S} = \frac{\underline{v}}{v}$$

$$\hat{T} = \hat{S} \times \hat{k}$$

$$\hat{R} = \hat{S} \times \hat{T}$$

where $\hat{k}$ is a unit vector in the direction of the Z axis in planetocentric ecliptic coordinates. The magnitude of the B vector is

$$|\underline{B}| = \sqrt{p\ |a|}$$

and the unit vector in the direction of $\underline{B}$ is

$$\hat{B} = \hat{S} \times \left(\frac{\underline{r} \times \underline{v}}{|\underline{r} \times \underline{v}|}\right)$$

therefore the B-plane parameters are

$$\underline{B} \cdot \hat{T} = (|\underline{B}|\,\hat{B}) \cdot \hat{T}$$

$$\underline{B} \cdot \hat{R} = (|\underline{B}|\,\hat{B}) \cdot \hat{R}$$

Two other important parameters to know are the time of closest approach and the time at the sphere of influence. The time at closest approach (t_{CA}) is

$$t_{CA} = t - \frac{M}{n}$$

where M is the value of the mean anomaly at t and n is the mean motion. The time at the sphere of influence (t_{SOI}) is computed from the following equations

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

$$\sinh F = \text{sign}\ (\underline{r} \cdot \underline{v})\ \sqrt{(\cosh F_1)^2 - 1}$$

$$\cosh F_{SOI} = \frac{1}{e}\left(1 - \frac{r_{SOI}}{a}\right)$$

$$\sinh F_{SOI} = \text{sign}\ (\underline{r} \cdot \underline{v})\ \sqrt{(\cosh F_{SOI})^2 - 1}$$

$$t_{SOI} = t + \frac{1}{n}\Big[(\sinh F_{SOI} - \sinh F)\ e$$

$$+ (F - F_{SOI})\Big]$$

For the case where closest approach is outside of the sphere of influence, the sphere of influence is assumed to be closest approach. All calculations are the same except

$$t_{SOI} = t_{CA}$$

When closest approach is outside the sphere of influence, the B-plane parameters are undefined; but closest approach parameters can be defined from the following equations.

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

$$e \cdot \sin E = \frac{\underline{r} \cdot \underline{v}}{\sqrt{\mu a}}$$

$$\tan E = \frac{\sin E}{\cos E}$$

$$M = E - e \sin E$$

$$t_{CA} = t - \frac{M}{n}$$

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
R	I	A	Position vector relative to the target body.
V	I	A	Velocity vector relative to the target body.
TO	I	A	Time associated with R and V.
BDT	O	C	$\underline{B} \cdot \underline{T}$.

Variable	Input/ Output	Argument/ Common	Definition
BDR	O	C	$\underline{B} \cdot \underline{R}$.
TSØI	O	C	Time at the sphere of influence, t_{SOI}.
NTP	I	A	Number of the target body. This flag is used to locate the SOI size and mass of the target body in the SPHERE and PMASS arrays.
VHP	O	C	Hyperbolic excess velocity, V_{hp}.
PI	I	C	3.14159....
PMASS	I	C	Array containing the masses of the planets.
SPHERE	I	C	Array containing the sphere sizes of the planets.
VCA	O	C	Velocity at closest approach.
RCA	O	C	Radius of closest approach.
TCA	O	C	Time of closest approach.
A	O	C	Semi-major axis of the osculating conic.
E	O	C	Eccentricity of the osculating conic.
XINC	O	C	Inclination of the osculating conic.
ØMEGA	O	C	Longitude of the ascending node.
SØMEGA	O	C	Argument of periapsis.
XMEAN	O	C	Mean anomaly.

Variable	Input/ Output	Argument/ Common	Definition
TA	O	C	True anomaly.
BIG	I	C	10^{30}.

Local Variables:

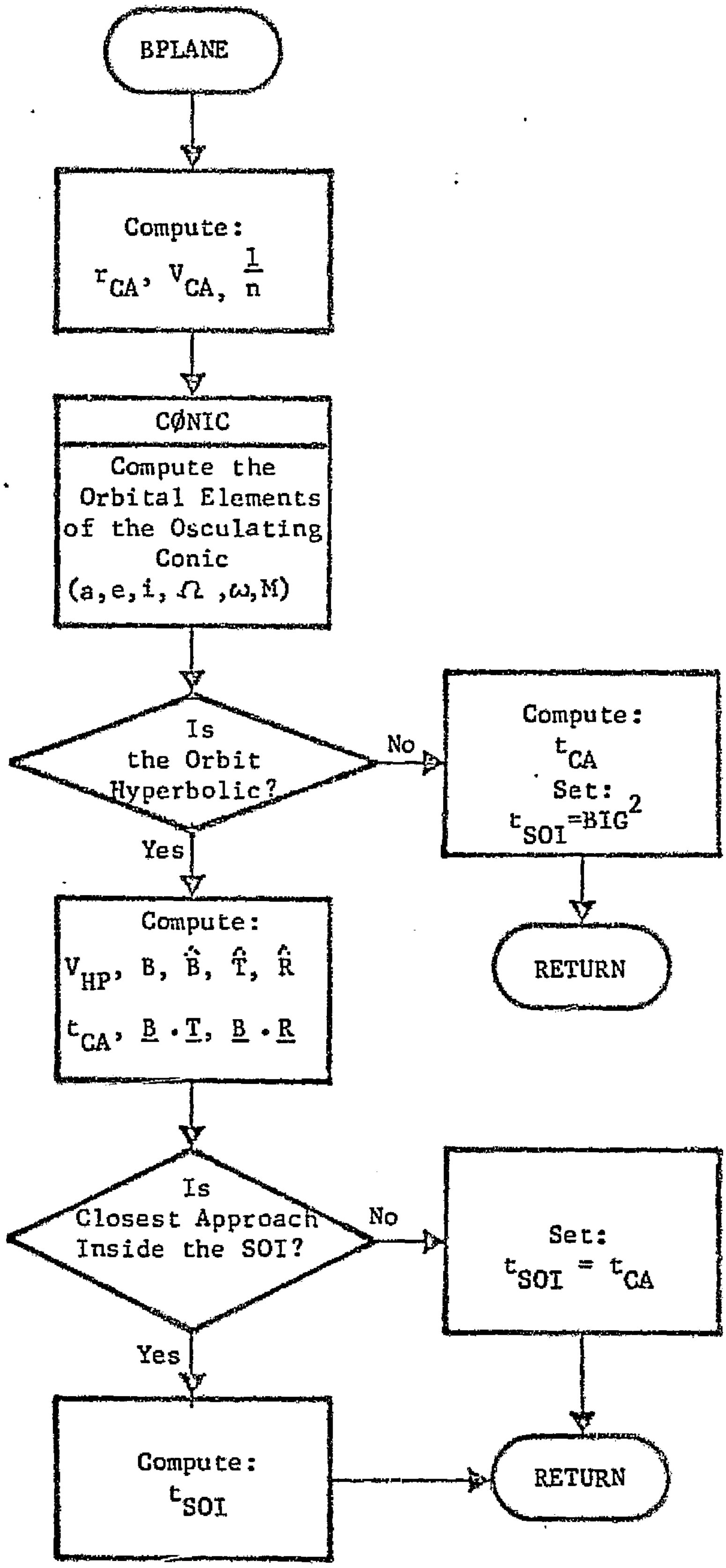
Variable	Definition
GMU	Mass of the target body.
RS	SOI size of the target body.
XN	Inverse of the mean motion, n
SV	$\hat{S}$
BV	$\hat{B}$
B	$\lvert \underline{B} \rvert$
TMAG	$\lvert \underline{S} \rvert$
RVX, RVY, RVZ	Components of $\hat{R}$
THETA	Angle between $\underline{B}$ and the $\hat{T}$ axis.
CØSHF1	cosh F
CØSH2	cosh F_{SOI}
SINF1	sinh F
SINF2	sinh F_{SOI}
F1	F
F2	F_{SOI}
DT	Time from the sphere to $\underline{r}$.
CE	cos E
SE	sin E
ECC	E
XM	Mean anomaly, M

<u>Subroutines Called:</u> CØNIC

<u>Calling Subroutines:</u> TCØMP, TREK

<u>Common Blocks:</u> CØNICS, CØNST, EPHEM, TARG

Logic Flow:


BPLANE
Compute:
r_{CA}, V_{CA}, $\frac{1}{n}$
CØNIC
Compute the Orbital Elements of the Osculating Conic
(a, e, i, Ω, ω, M)
Is the Orbit Hyperbolic?
No
Compute:
t_{CA}
Set:
t_{SOI}=BIG^2
RETURN
Yes
Compute:
V_{HP}, B, $\hat{B}$, $\hat{T}$, $\hat{R}$
t_{CA}, $\underline{B} \cdot \underline{T}$, $\underline{B} \cdot \underline{R}$
Is Closest Approach Inside the SOI?
No
Set:
$t_{SOI} = t_{CA}$
Yes
Compute:
t_{SOI}
RETURN

3.6.3 Subroutine: CARTES (A, E, XI, Ø, W, XM, GMU, R, V)

Purpose: To compute the cartesian state vector corresponding to a set of orbital elements at a given time. Time is implicit in the Mean Anomaly XM.

Method: Conic Formulae for Elliptic and Hyperbolic Motion.

Input/Output:

Variable	I/O	Argument/ Common	Definition
A	I	A	Semi-major Axis (a)
E	I	A	Eccentricity (e)
XI	I	A	Inclination (i)
Ø	I	A	Longitude of the Ascending Node (Ω)
W	I	A	Argument of Periapsis (ω)
GMU	I	A	Gravitational Constant (μ)
R	O	A	Position Vector ($\underline{r}$)
V	O	A	Velocity Vector ($\underline{v}$)
PI	I	C	3.14159...........
XM	I	A	Mean Anomaly (M)

Local Variables:

Variable	Definition
ITT	Iteration counter for Kepler's Equation
NITT	Maximum iterations for Kepler's Equation
FP	Derivative of Kepler's Equation ($f'(x_n)$)

Variable	Definition
ECC	Eccentric Anomaly (x_n)
FN	Kepler's Equation $(f(x_n'))$
SQE	$\sqrt{\frac{1+E}{\lvert 1-E \rvert}}$
TA	True Anomaly
RM	Magnitude of the Position Vector
SINHE	XM/E
CØSHE	$\sqrt{1+SINHE^2}$
SINHEC	Hyperbolic Sine of ECC
CØSHEC	Hyperbolic Cosine of ECC
P	Semi-latus Rectum
TH	Argment of Latitude
CØSTH	Cosine of TH
SINTH	Sine of TH
CØSØ	Cosine of ϕ
SINØ	Sine of ϕ
CØSW	Cosine of W
CØSI	Cosine of XI
SINI	Sine of XI
VA	$\sqrt{GMU/P}$
VB	SINTH + E * SINW
VC	CØSTH + E * CØSW

Remarks:

Given: The orbital elements a, e, i, Ω, ω and the gravitational constant μ.

Find: The position $\underline{r}$ and the velocity $\underline{v}$.

First we must find the eccentric anomaly E for the elliptical case and H for the hyperbolic in terms of M, the mean anomaly. For the elliptical case

$$M = E - e \cdot \sin E$$

and for the hyperbolic case

$$M = e \cdot \sinh H - H$$

Since both equations are transcendental we must solve them interatively. The method used to solve these equations is Newton's Method of the form

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

Therefore, for the elliptical case the expression is

$$E_{n+1} = E_n - \frac{E_n - e \cdot \sin E_n - M}{1 - e \cdot \cos E_n}$$

and for the hyperbolic case the expression is

$$H_{n+1} = H_n - \frac{e \cdot \sinh H_n - H_n - M}{e \cdot \cosh H_n - 1}$$

Depending on the kind of orbit defined by the orbital elements, the appropriate equation is iterated upon until

$$|f(x)| \leq 10^{-10}$$

or

$$\left|\frac{f(x)}{f'(x)}\right| \leq 10^{-10}$$

for a finite number of iterations.

Now that we have E or H we can find r and v from the following equations:

Elliptical

$$\tan\left(\frac{f}{2}\right) = \left(\frac{1+e}{1-e}\right) \cdot \tan\left(\frac{E}{2}\right)$$

$$r = a\ (1-e\cdot\cos E)$$

$$p = a\ (1-e^2)$$

Hyperbolic

$$\tan\left(\frac{f}{2}\right) = \left(\frac{e+1}{e-1}\right) \cdot \tanh\left(\frac{H}{2}\right)$$

$$r = a\ (1-e\cdot\cosh H)$$

$$p = a\ (e^2-1)$$

$$\underline{r} = r \begin{bmatrix} \cos\Omega\cdot\cos\theta - \sin\Omega\cdot\sin\theta\cdot\cos i \\ \sin\Omega\cdot\cos\theta + \cos\Omega\cdot\sin\theta\cdot\cos i \\ \sin\theta\cdot\sin i \end{bmatrix} \qquad \text{where } \theta = \omega + f$$

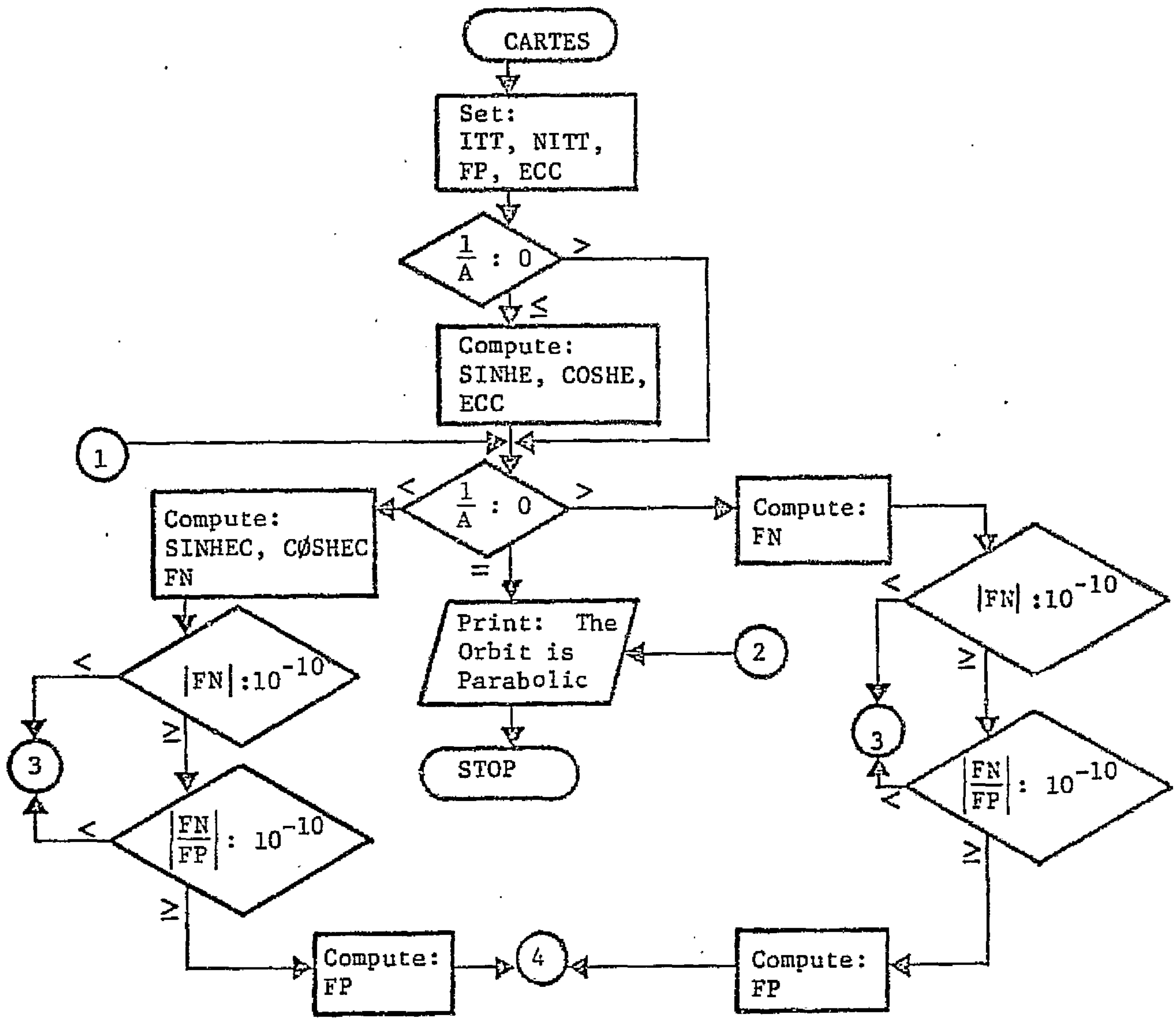
$$\underline{v} = \sqrt{\frac{\mu}{p}} \begin{bmatrix} \cos\Omega(\sin\theta + e\cdot\sin\omega) + \sin\Omega\cdot\cos i\ (\cos\theta + e\cdot\cos\omega) \\ \sin\Omega(\sin\theta + e\cdot\sin\omega) - \cos\Omega\cdot\cos i\ (\cos\theta + e\cdot\cos\omega) \\ -(\cos\theta + e\cdot\cos\omega)\cdot\sin i \end{bmatrix}$$

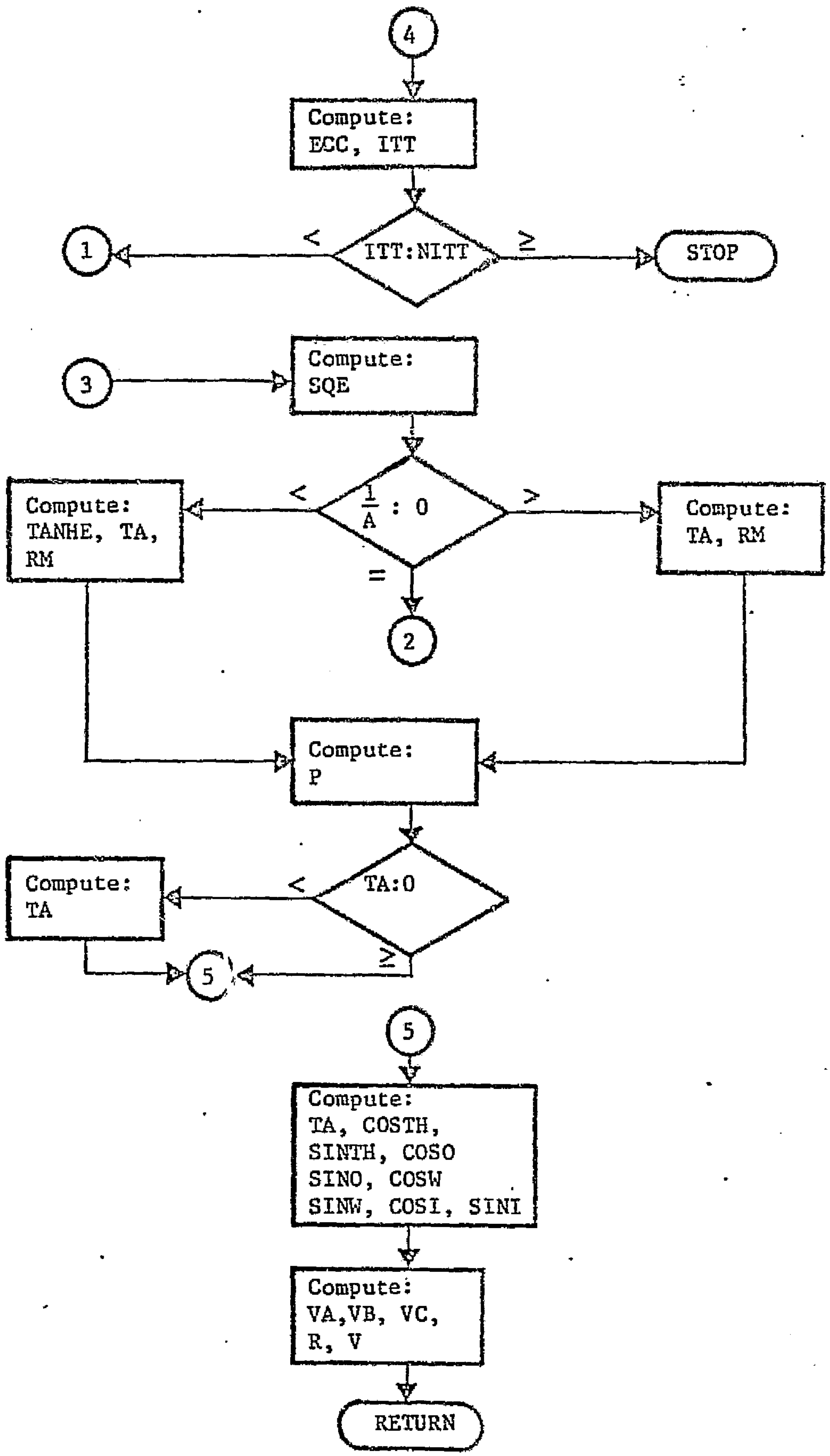
Subroutines Called: None

Calling Program: EPHEM

Common Block: CØNST

Logic Flow:




4
Compute:
ECC, ITT
ITT:NITT
<
1
≥
STOP
3
Compute:
SQE
1/A : 0
<
Compute:
TANHE, TA,
RM
>
Compute:
TA, RM
=
2
Compute:
P
TA:0
<
Compute:
TA
≥
5
5
Compute:
TA, COSTH,
SINTH, COSO
SINO, COSW
SINW, COSI, SINI
Compute:
VA,VB, VC,
R, V
RETURN

3.6.4 Subroutine: CØNIC (R, V, GMU, A, E, XINC, ØMEGA, SØMEGA, XMEAN, THETA)

Purpose: To compute the orbit elements given a state vector and the corresponding time.

Method: Conic Formulae for Elliptic and Hyperbolic motion.

Remarks:

Given: The position vector $\underline{r}$, the velocity vector $\underline{v}$ and the gravitational constant μ.

Find: The orbital elements a, e, i, Ω ω and M and also θ

$$\underline{h} = \underline{r} \times \underline{v}$$

$$\underline{w} = \underline{h}/h$$

$$r_v = \underline{r} \cdot \underline{v}$$

$$\underline{e} = \frac{1}{\mu} (\underline{v} \times \underline{h}) - \underline{r}/r$$

$$p = h/\mu$$

$$\alpha = (\frac{2}{r} - \frac{v^2}{\mu})$$

$$\underline{\rho} = \underline{e}/e$$

$$\underline{q} = \underline{w} \times \underline{\rho}$$

$$\sin\theta = \frac{h \cdot r_v}{r}$$

$$\cos\theta = \frac{h^2 - \mu}{r}$$

Now

$$a = \frac{1}{\alpha}$$

$$e = |\underline{e}|$$

$$i = \cos^{-1} (w_z)$$

$$\Omega = \tan^{-1} (w_x/-w_y)$$

$$\omega = \tan^{-1} (\rho_z/q_z)$$

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

$$\cos E = 1 - r\cdot\alpha$$

$$\sin E = \frac{r_v \cdot \alpha}{\mu}$$

for the elliptical case

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

$$M = E - e\cdot\sin E$$

for the hyperbolic case

$$\sinh H - \sin E/e$$

$$\cosh H = \cos E/e$$

$$H = \ln(\sinh H - \cosh H)$$

$$M = e\cdot\sinh H - H$$

Input/Output:

Variable	I/O	Argument/ Common	Definition
R	I	A	Position Vector ($\underline{r}$)
V	I	A	Velocity Vector ($\underline{v}$)
TO	I	A	Time Corresponding to $\underline{r}$ and $\underline{v}$
GMU	I	A	Gravitational Constant (μ)
A	O	A	Semi-Major Axis (a)
E	O	A	Eccentricity (e)
XINC	O	A	Inclination of the orbit plane (i)

Variable	I/O	Argument/ Common	Definition
OMEGA	O	A	Longitude of the Ascending Node (Ω)
SOMEGA	O	A	Argument of the Periapsis (ω)
XMEAN	O	A	Mean Anomaly
PI	I	C	3.14159.....

Local Variables:

Variable	Definition
H	Magnitude of the Angular Momentum Vector (h)
HV	Angular Momentum Vector ($\underline{h}$)
WV	Unit Vector in the direction of ($\underline{w}$)
RM	Magnitude of $\underline{r}$
VM	Magnitude of $\underline{v}$
RDV	$\underline{r} \cdot \underline{v}$
EV	$\underline{e}$
AA	$\frac{1}{a}$
P	Semi-Latus Rectum
PV	$\underline{e}/e$
QV	$\underline{h} \times (\underline{e}/e)$
THETA	Argument of latitude
STH	Sine of THETA
CTH	Cosine of THETA
ECC	Eccentric Anomaly
CE	Sine of ECC

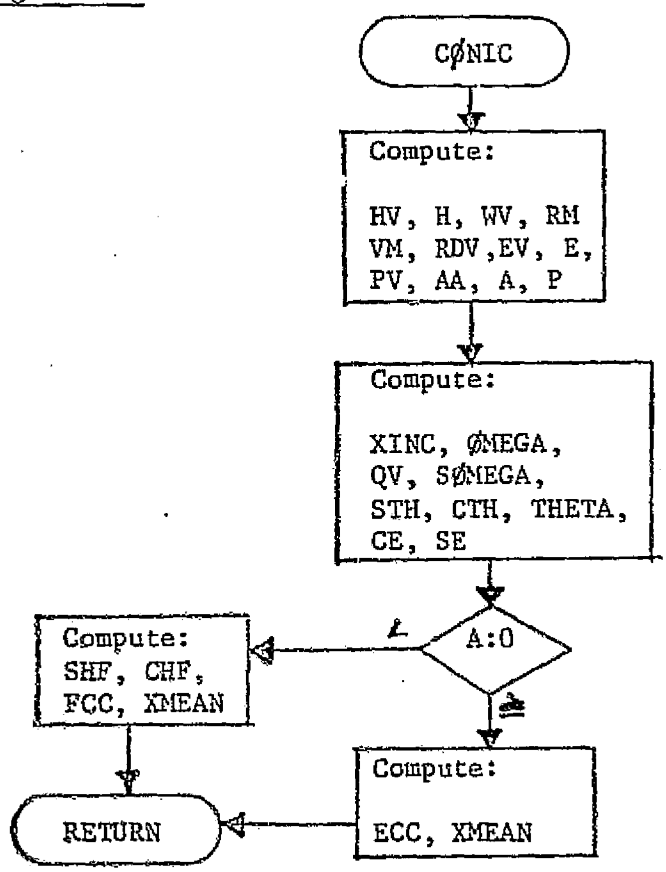
Variable	Definition
SE	Cosine of ECC
FCC	Hyperbolic Anomaly
CHF	Hyperbolic Sine of FCC
SHF	Hyperbolic Cosine of FCC

Subroutines Called: UXV, VECMAG, UNITV, UDØTV

Calling Subroutines: BPLANE, PRØP, EPHERR, ØD, PGM, DATAT, FEGS

Common Blocks: CØNICS, CØNST

Logic Flow:



3.6.5 <u>Subroutine</u>: ECØMP (XX, VV, TSTØP, NTARG, NTP, LISTAR, ETA)

<u>Purpose</u>: To compute the transformation matrix which transforms state vector deviations into target variable deviations at the target time; namely

$$\eta = \left[\frac{\partial(T_1, T_2, \ldots, T_m)}{\partial(X, Y, Z, \dot{X}, \dot{Y}, \dot{Z})} \right]_{(m \times 6)} ,$$

where m is the number of target variables.

<u>Method</u>: Small changes to the trajectory state vector at the target time permit this transformation matrix to be computed by numerical differencing. Central difference partial derivatives are used.

<u>Remarks</u>: Currently, the state vector deviations used to generate the numerical partials are 10 km for position and 10 m/sec for velocity. For some applications, in particular for missions to the inner planets (Mercury and Venus), these values may have to be reduced.

<u>Input/Output</u>:

Variable	Input/ Output	Argument/ Common	Definition
XX	I	A	State vector position components.
VV	I	A	State vector velocity components.

Variable	Input/ Output	Argument/ Common	Definition
TSTØP	I	A	Epoch of state vector evaluation; generally the target time.
LISTAR	I	A	List of target variable codes to be passed to TCØMP.
NTARG	I	A	Number of target variables.
NTP	I	A	Target planet number.
ETA	O	A	η - matrix of partial derivatives.

Local Variables:

Variable	Definition
DEL	State vector perturbations.
XT1	Backward step target variables.
XT2	Forward step target variables.

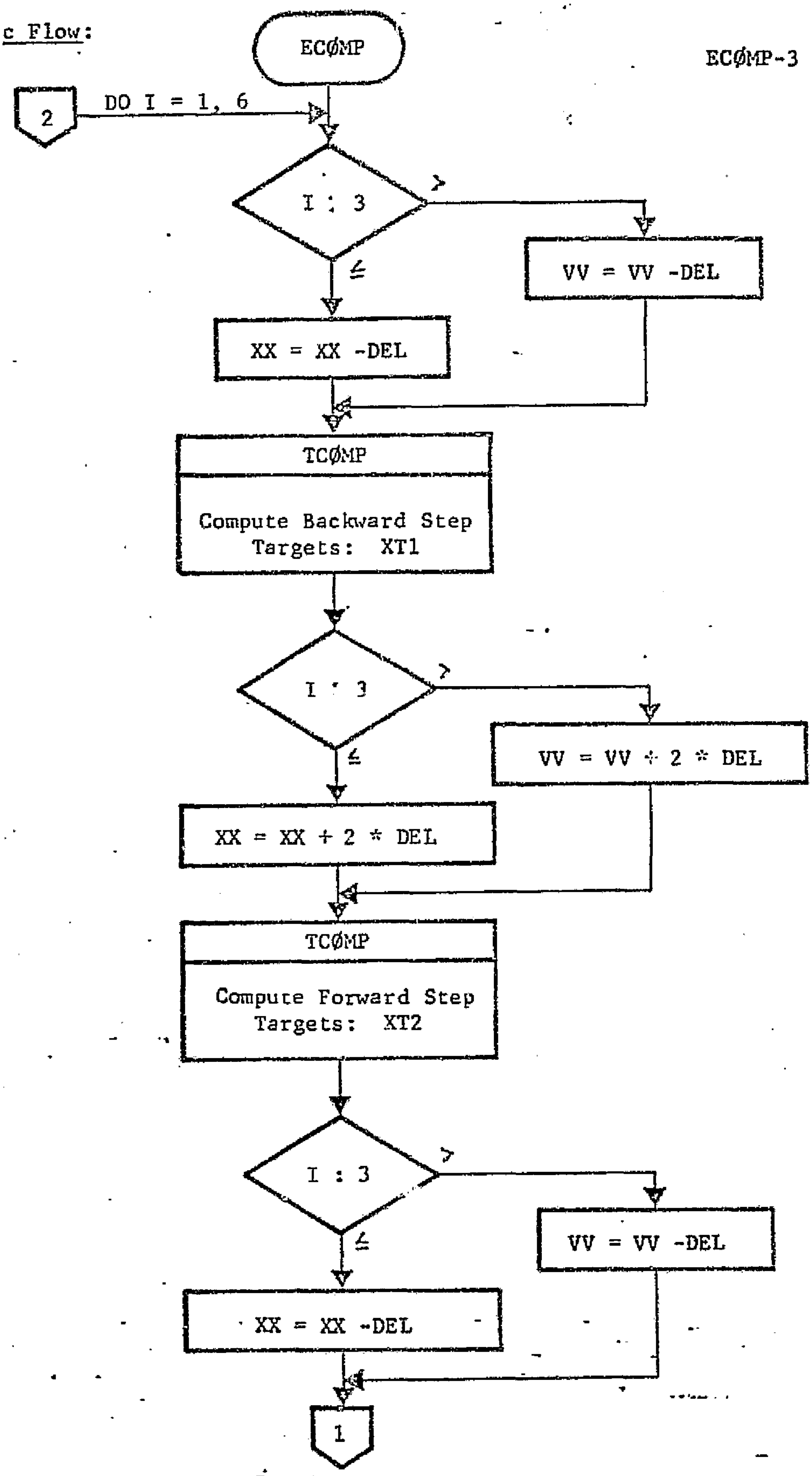
Subroutines Called: TCØMP

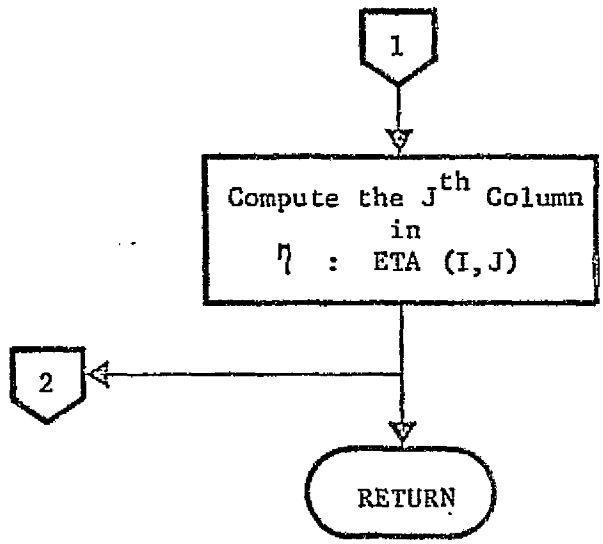
Calling Subroutines: LGUID, NGUID, GUIDE, NLGUID, REFTRJ, STMTAR

Common Blocks: WØRK

Logic Flow:

Logic Flow:


ECØMP
2
DO I = 1, 6
I : 3
>
≤
VV = VV -DEL
XX = XX -DEL
TCØMP
Compute Backward Step Targets: XT1
I : 3
>
≤
VV = VV + 2 * DEL
XX = XX + 2 * DEL
TCØMP
Compute Forward Step Targets: XT2
I : 3
>
≤
VV = VV -DEL
XX = XX -DEL
1


1
Compute the J th Column in
η : ETA (I,J)
2
RETURN

3.6.6 Subroutine: ENCØN (T)

Entry Points: REFINE, ØSCUL

Purpose: To propagate the reference conic from rectification to time t.

Method: Conic equations for elliptical and hyperbolic orbits. See MAPSEP Analytic Manual (Reference 1), Appendix 1 (Section 9.1).

Remarks: Common block ENCØN contains local variables (denoted by asterisk) to save these values when GODSEP (using the PDOT option) temporarily replaces the TRAJ overlay with the MEAS overlay. Thus, CØMMØN/ENCØN/ is required only for a very specific application.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
T	I	A	Trajectory time in seconds
TSTØP	I	C	The sign of TSTØP determines whether the propagation is backwards (-) or forwards (+).
NPRI	I	C	A flag that is used to locate the mass of the primary body in the PMASS array.
PMASS	I	C	Array containing the masses of all the bodies.
ALPHA	Ø	C	Inverse semi-major axis ($\frac{1}{a}$).
UTRUE	I	C	Position vector at rectification (r_o).
VTRUE	I	C	Velocity vector at rectification (V_o).
UENC	Ø	C	Osculating conic position vector at time t.
UENCM	Ø	C	Magnitude of UENC.
VENC	Ø	C	Osculating conic velocity vector at time t.
VENCM	Ø	C	Magnitude of VENC.

Local Variables:

Variable	Definition
* TZERØ	Time of rectification (t_o).
GMU	Mass of the reference body.
* UZERØ	Position vector at t_o, ($\underline{r}_o$).
* VZERØ	Velocity vector at t_o, ($\dot{\underline{r}}_o$).
CZERØ	1 + e cos E_o for the elliptical case.
	1 + e cosh H_o for the hyperbolic case.
UALPHA	1 - e cos E_o for the elliptical case.
	e cosh H_o - 1 for the hyperbolic case.
* UBETA	Absolute value of UALPHA.
BETA	Absolute value of ALPHA.
* A1	Mean angular motion (n).
* A2	e sin E_o for the elliptical case.
	e sin H_o for the hyperbolic case.
* A3	e cos E_o for the elliptical case.
	e cosh H_o for the hyperbolic case.
C1	e exp $[H_o]$ for the hyperbolic case.
C2	e exp $[-H_o]$ for the hyperbolic case.
* DELE	$E - E_o$ for the elliptical case.
* X	exp $[H - H_o]$ -1 for the hyperbolic case.
HV	The angular momentum vector ($\overline{r}_o \times \overline{v}_o$).
ARG1	$1 - \frac{a}{r_o}\left[1 - \cos(E - E_o)\right]$ for the elliptical case.

* actually contained in CØMMØN/ENCØN/

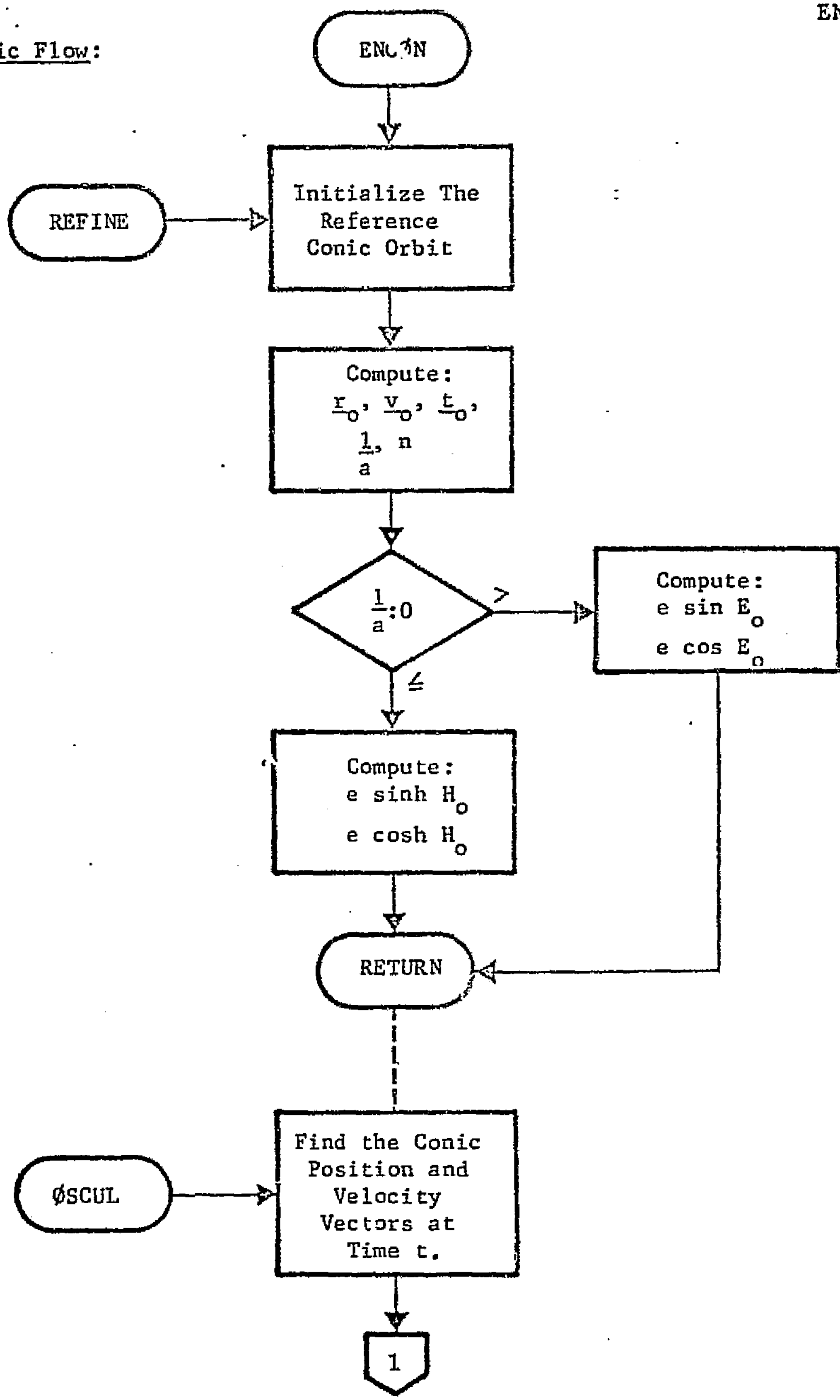
Variable	Definition
	$1 - \frac{a}{r_o}\left[\cosh(H - H_o) - 1\right]$ for the hyperbolic case.
ARG2	$\frac{1}{n}\left[\sin(E - E_o) - e(\sin E - \sin E_o)\right]$ for the elliptical case.
	$\frac{1}{n}\left[e(\sinh H - \sinh H_o) - \sinh(H - H_o)\right]$ for the hyperbolic case.
ARG3	$-\frac{\sqrt{\mu a}}{r r_o}\sin(E - E_o)$ for the elliptical case.
	$-\frac{\sqrt{\mu a}}{r r_o}\sinh(H - H_o)$ for the hyperbolic case.
ARG4	$1 - \frac{a}{r}\left[1 - \cos(E - E_o)\right]$ for the elliptical case.
	$1 - \frac{a}{r}\left[\cosh(H - H_o) - 1\right]$ for the hyperbolic case.

Subroutines Called: VECMAG, UXV, UDØTV

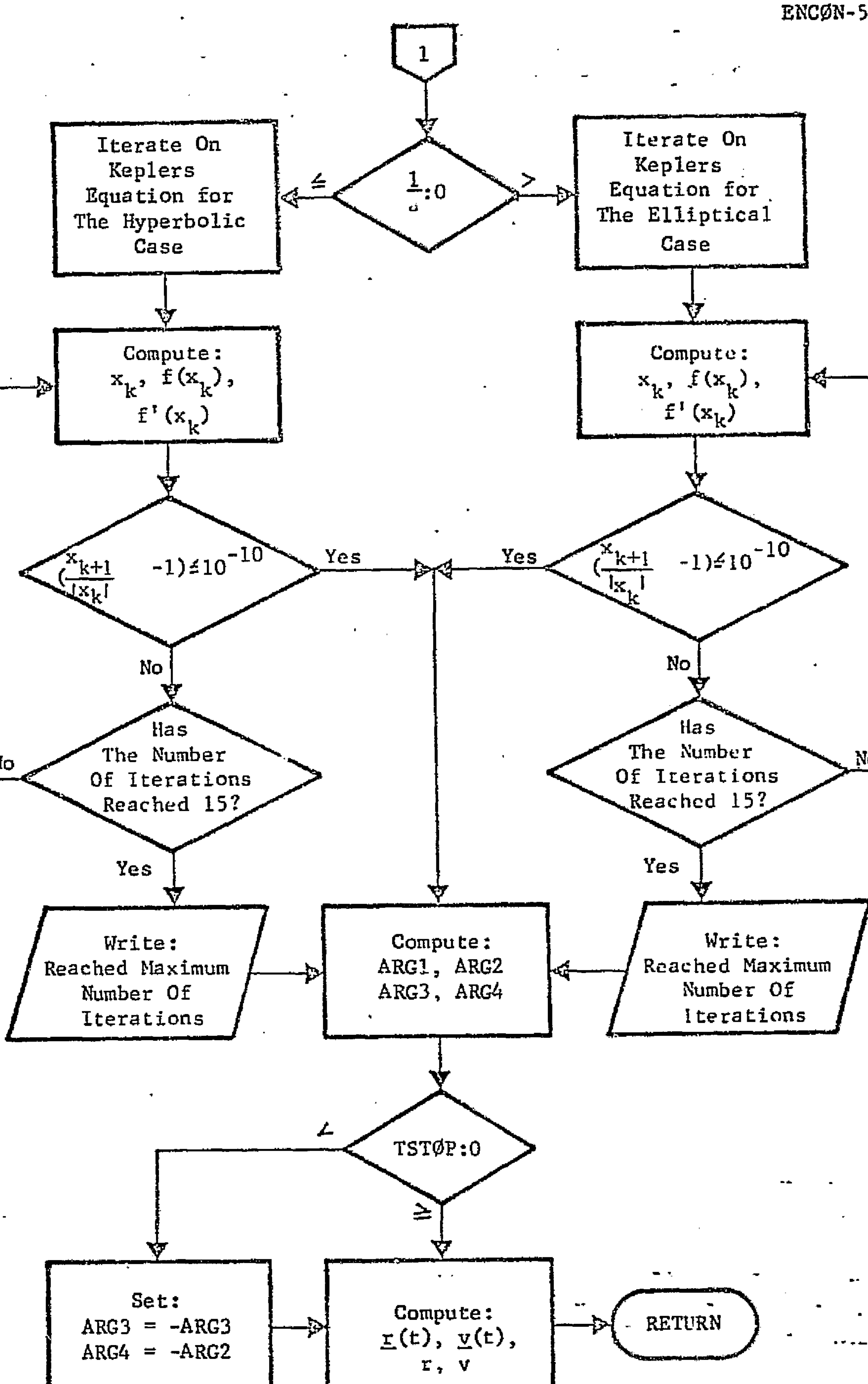
Calling Subroutines: MØTIØN

Common Blocks: ENCØN, EPHEM, TIME, TRAJ1, TRAJ2

Logic Flow:



ENCØN
REFINE
Initialize The Reference Conic Orbit
Compute: $\underline{r}_o$, $\underline{v}_o$, $\underline{t}_o$, $\frac{1}{a}$, n
$\frac{1}{a}$:0
>
≤
Compute: e sin E_o e cos E_o
Compute: e sinh H_o e cosh H_o
RETURN
ØSCUL
Find the Conic Position and Velocity Vectors at Time t.
1


1
1/a:0
≤
>
Iterate On Keplers Equation for The Hyperbolic Case
Iterate On Keplers Equation for The Elliptical Case
Compute: x_k, f(x_k), f'(x_k)
Compute: x_k, f(x_k), f'(x_k)
(x_k+1/|x_k| -1)≤10^-10
Yes
Yes
(x_k+1/|x_k| -1)≤10^-10
No
No
Has The Number Of Iterations Reached 15?
No
Has The Number Of Iterations Reached 15?
No
Yes
Yes
Write: Reached Maximum Number Of Iterations
Compute: ARG1, ARG2 ARG3, ARG4
Write: Reached Maximum Number Of Iterations
TSTØP:0
<
≥
Set: ARG3 = -ARG3 ARG4 = -ARG2
Compute: r(t), v(t), r, v
RETURN

3.6.7 Subroutine: GENINV (A, M, N, B)

Purpose: To compute an inverse B for any m x n matrix A.

Remarks: There are three cases for which GENINV will compute an inverse.

Case 1: $m < n$

$$B = A^T[A A^T]^{-1}$$

Case 2: $m = n$

$$B = A^{-1}$$

Case 3: $m > n$

$$B = [A^T A]^{-1} A^T$$

The matrices A and B can share the same location only if m = n.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
A	I	A	The matrix to be inverted.
M	I	A	Number of rows in A (Columns in B).
N			Number of columns in A (Rows in B).
B	I	A	Inverse of A.

Local Variables:

Variable	Definition
WØRK	Array used for temporary calculations.

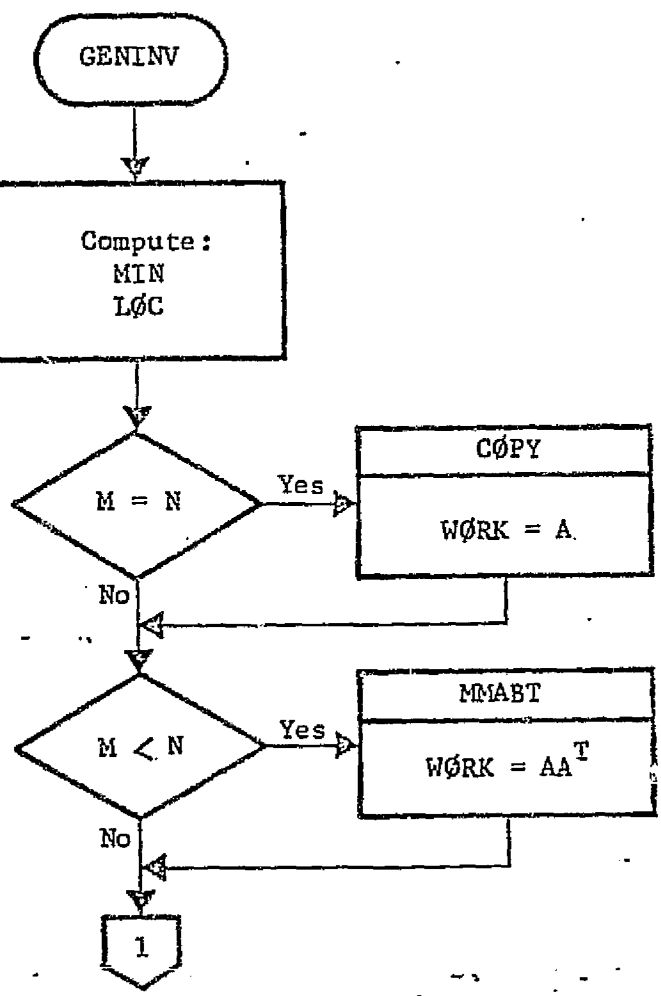
Variable	Definition
MIN	Number of needed locations for temporary calculations.
LØC	Number of needed locations for the inverse.

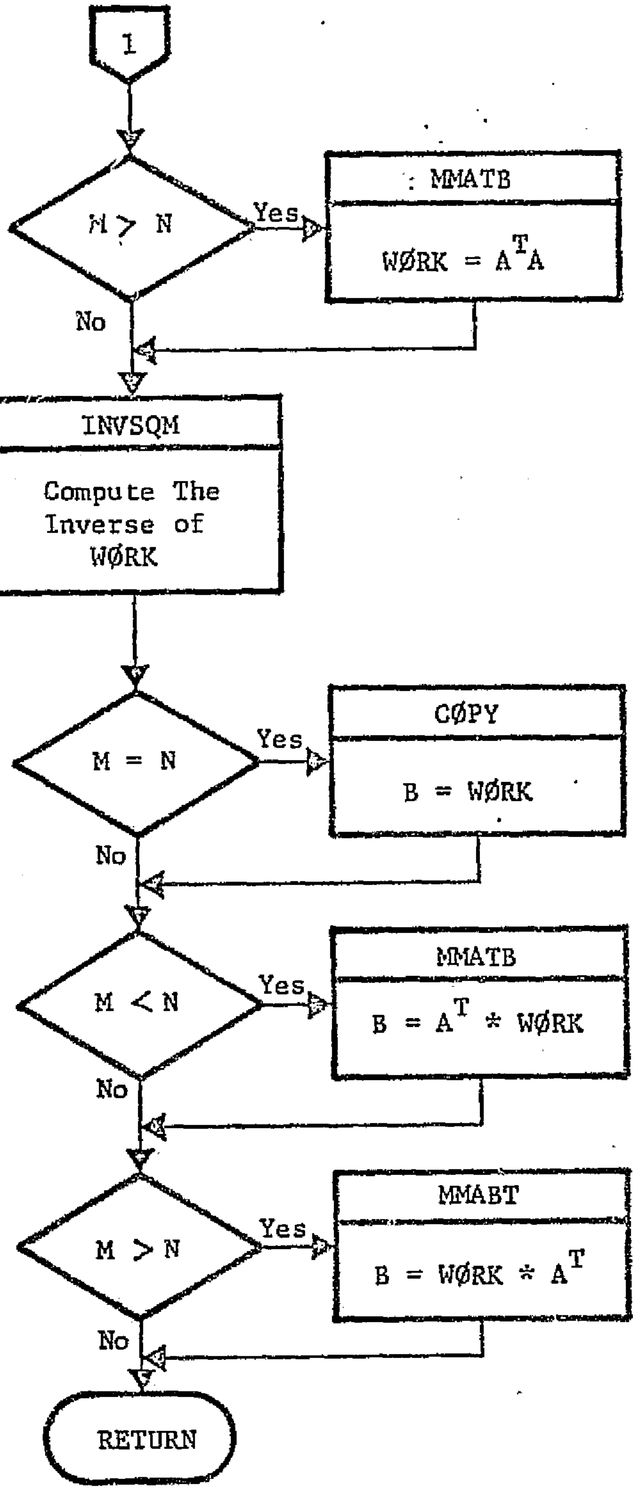
Subroutines Called: CØPY, MMABT, MMATB, INVSQM

Calling Subroutines: GUIDE, LGUID, NLGUID

Common Blocks: WØRK

Logic Flow:




1
M > N
Yes
No
MMATB
WØRK = A^T A
INVSQM
Compute The Inverse of WØRK
M = N
Yes
No
CØPY
B = WØRK
M < N
Yes
No
MMATB
B = A^T * WØRK
M > N
Yes
No
MMABT
B = WØRK * A^T
RETURN

3.6.8 Subroutine: MPAK (A, M, N, ASUB, MSUB, NSUB)

Purpose: MPAK is used to (1) copy subblocks of matrix A into a matrix ASUB, (2) copy the diagonal elements of matrix A into ASUB which can be a vector (or row matrix) or (3) "pack" the matrix A. M and N are the dimensions of A, and MSUB and NSUB are the dimensions of ASUB.

Method: An mxn matrix is stored internally in the computer by columns. Take the 3 x 3 matrix

$$E = \begin{bmatrix} e_{11} & e_{12} & e_{13} \\ e_{21} & e_{22} & e_{23} \\ e_{31} & e_{32} & e_{33} \end{bmatrix}$$

In the computer, E is stored as

Column 1	e_{11}
	e_{21}
	e_{31}
Column 2	e_{12}
	e_{22}
	e_{23}
Column 3	e_{13}
	e_{23}
	e_{33}

MPAK uses this information to perform one of the three following cases, (1) to copy sub blocks of E, (2) to copy the diagonal elements of E, and (3) to pack E.

Case 1: Given a 3 x 3 matrix

$$E = \begin{bmatrix} e_{11} & e_{12} & e_{13} \\ e_{21} & e_{22} & e_{23} \\ e_{31} & e_{32} & e_{33} \end{bmatrix}$$

copy the sub block

$$F = \begin{bmatrix} e_{21} & e_{22} \\ e_{31} & e_{32} \end{bmatrix}.$$

into the 2 x 2 matrix F. In order to accomplish this, MPAK must know the first element of the sub block to be copied. For this problem, it is e_{21}. The FORTRAN call to MPAK must transmit this information. Such a call would be

```
CALL MPAK (E(2,1), 3, 3, F, 2, 2)
```

Case 2: Given a 2 x 2 matrix

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$$

copy the diagonal terms a_{11} and a_{22} into the 2 x 1 row vector B. The call to MPAK is

```
CALL MPAK (A, 3, 2, B, 1, 2)
```

The dimension of A is given as 3 x 2. Internally in the computer, A is thought of as being stored

Column 1	a_{11}
	a_{21}
	a_{12}
Column 2	a_{22}

This particular call makes MPAK copy the elements a_{11} and a_{22} into B.

<u>Case 3</u>: Given the 3 x 3 matrix

$$A = \begin{bmatrix} a & c & o \\ b & d & o \\ o & o & o \end{bmatrix}$$

pack it so that

$$A = \begin{bmatrix} a & d & o \\ b & o & o \\ c & o & o \end{bmatrix}$$

Pack as used here, means to order the nonzero elements of A into consecutive locations internally. If

$$A = \begin{bmatrix} a & c & e & o \\ b & d & f & o \\ o & o & o & o \\ o & o & o & o \end{bmatrix}$$

than packing A would result in

$$A = \begin{bmatrix} a & e & o & o \\ b & f & o & o \\ c & o & o & o \\ d & o & o & o \end{bmatrix}$$

The appropriate call to MPAK would be

CALL MPAK (A, 3, 3, A, 2, 2)

for the first example (3 x 3 A), and for the second example:

CALL MPAK (A, 4, 4, A, 3, 3)

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
A	I	A	The matrix to be operated on
M	I	A	The number of rows of A
N	I	A	The number of columns of A
ASUB	O	A	The resultant matrix
MSUB	I	A	The number of rows of ASUB
NSUB	I	A	The number of columns of ASUB

Local Variables: None

Subroutines Called: None

Calling Subroutines: SIZE, SDAT, (GODSEP, et al)

Common Blocks: None

3.6.9 Subroutine: MUNPAK (ASUB, MSUB, NSUB, A, M, N)

Purpose: MUNPAK is used to copy a matrix ASUB into a large matrix A, to copy a row matrix ASUB onto the diagonal of A or to "unpack" the matrix ASUB.

Method: MUNPAK, like MPAK takes advantage of the way a matrix is stored internally in a computer. MUNPAK performs the reverse function of MPAK: (1) copy a matrix into a larger matrix, (2) copy a row matrix onto the diagonal of a matrix or (3) unpack the matrix.

Case 1: Copy a 2x2 matrix

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$$

into a 3x3 matrix B so that

$$B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & a_{11} & a_{12} \\ 0 & a_{21} & a_{22} \end{bmatrix}$$

This is accomplished by specifying where the first element of A is to be located in B. The FORTRAN call to MUNPAK is

CALL MUNPAK (A, 2, 2, B (2, 2), 3, 3)

Case 2: Copy the 1x2 row matrix

$$A = \begin{bmatrix} a_{11} & a_{12} \end{bmatrix}$$

into the 2x2 matrix B. In the call to MUNPAK, the dimensions of B are given as a 3x2. The net result is

$$B = \begin{bmatrix} a_{11} & 0 \\ 0 & a_{12} \end{bmatrix}$$

The call to MUNPAK is

CALL MUNPAK (A, 1, 2, B, 3, 2).

Case 3: Given the 3x3 matrix

$$A = \begin{bmatrix} a & d & 0 \\ b & 0 & 0 \\ c & 0 & 0 \end{bmatrix}$$

"unpack" it so that

$$A = \begin{bmatrix} a & c & 0 \\ b & d & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

The call to MUNPAK to accomplish this operation is

CALL MUNPAK (A, 2, 2, A, 3, 3).

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
ASUB	I	A	The matrix to be operated on.
MSUB	I	A	The number of rows of ASUB.
NSUB	I	A	The number of columns of ASUB.
A	O	A	The resultant matrix.
M	I	A	The number of rows of A.
N	I	A	The number of columns of A.

Local Variables: None

Subroutines Called: None

Calling Subroutines: SIZE, SDAT, (GØDSEP, et al.)

Common Blocks: None

3.6.10 Function: RNUM (SIGMA, IRAN)

Purpose: To sample a uniform distribution and generate random samples on a Gaussian distribution.

Method: Two random samples from a uniform distribution are made to form a random sample on a zero-mean, Gaussian distribution which has a unit standard deviation. The random variable on the Gaussian distribution is scaled according to the input standard deviation, SIGMA. For IRAN equal to zero, a one-sigma, forced Monte Carlo sample is computed and returned.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
SIGMA	I	A	Standard deviation of the random variable being sampled.
IRAN	I	A	Flag to indicate whether or not a forced Monte Carlo sample is to be returned.
RNUM	O	A	Resultant random variable.

Local Variables:

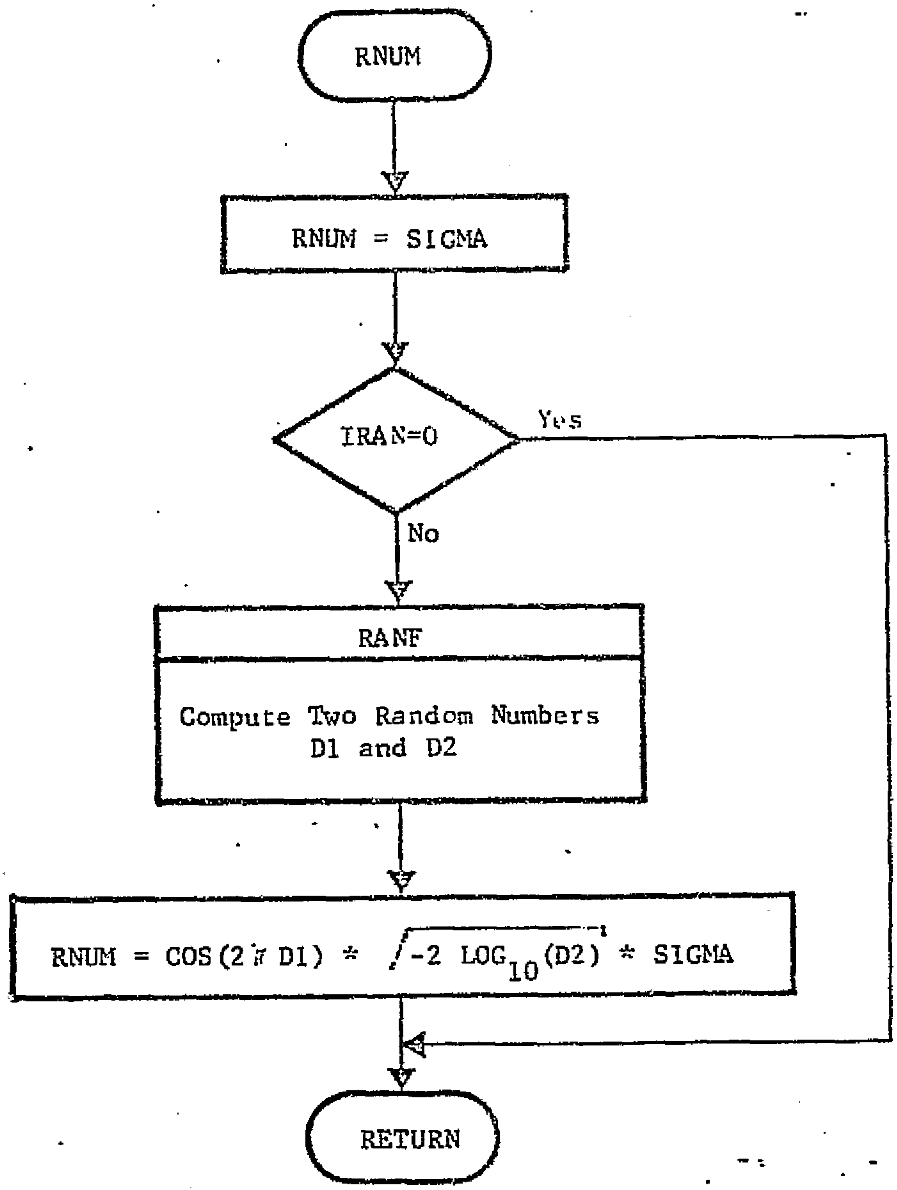
Variable	Definition
D1	First random sample from a uniform distribution.
D2	Second random sample from a uniform distribution.

Subroutines Called: RANF

<u>Calling Subroutines:</u> CSAMP, EXGUID, EPHSMP, ERRSMP, DNØISE

<u>Common Blocks:</u> CØNST

<u>Logic Flow:</u>



3.6.11 **Subroutine:** TCØMP (XX, VV, TSTØP, NTP, NTARG, LESTAR, XTARG, IPASS)

Entry: TCØMP1

Method: The BPLANE utility routine is called to compute osculating values of target variables corresponding to a given state vector. Individual target values are loaded into a target vector according to the target codes in the LISTAR array.

Input/Output:

Variable	Input/ Output	Argument Common	Defirition
XX	I	A	State vector position components
VV	I	A	State vector velocity components
TSTØP	I	A	Epcch corresponding to the state vector; generally the target time.
NTP	I	A	Number of the target planet.
NTARG	I	A	Number of target variables.
LISTAR	I	A	List of target variable codes.
XTARG	O	A	Target vector.
IPASS	I	A	Flag to control logic transfer.
VHP	I	C	Hyperbolic excess velocity.
RCA	I	C	Radius of closest approach.
BDT	I	C	T-coordinate in the B-plane.
BDR	I	C	R-coordinate in the B-plane.
TSØI	I	C	Conically interpolated time of arrival at the sphere of influence.

Variable	Input/ Output	Argument Common	Definition
TCA	I	C	Conically interpolated time of arrival at the radius of closest approach.
A	I	C	Semi-major axis evaluated on an osculating conic.
E	I	C	Eccentricity evaluated on an osculating conic.
XINC	I	C	Inclination evaluated on an osculating conic.
ØMEGA	I	C	Argument of the ascending node evaluated on an osculating conic.
SØMEGA	I	C	Argument of periapsis evaluated on an osculating conic.
XMEAN	I	C	Mean anomaly evaluated on an osculating conic.
TA	I	C	True anomaly evaluated on an osculating conic

Local Variables: None

Subroutine Called: BPLANE, VECMAG

Calling Subroutines: ECØMP, NLGUID, REFTRJ, SIMSEP, STMTAR, TREK

Common Blocks: CØNST, TARGET

Logic Flow: See Listing

Page 553 has been deleted.

3.6.12 Subroutine: THCØMP (XIN, MIN, NPRIN, NATC, LJU, TGØ, THALT, IMAN, XØUT, MØUT, THETA, PHI)

Purpose: To complete the $\hat{\Theta}_u$ and $\bar{\Phi}$ matrices which are used for trajectory targeting over a specified trajectory arc.

Method: THCØMP computes and stores certain partitions of the augmented state transition matrix into the $\hat{\Theta}_u$ and $\bar{\Phi}$ matrices as outlined in Appendix 7 of the Analytic Manual.

Remarks: This routine is used by TØPSEP and SIMSEP for evaluating $\hat{\Theta}_u$ and $\bar{\Phi}$. TØPSEP also has an alternate set of logic which uses a numerical differencing algorithim for the same purpose. SIMSEP uses THCØMP exclusively.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
XIN	I	A	Initial state vector.
MIN	I	A	Initial S/C mass.
NPRIN	I	A	Primary body code to which XIN is referenced.
NATC	I	A	Number of active thrust controls.
LJU	I	A	Array of active thrust control codes.
TGØ	I	A	Initial trajectory time.

Variable	Input/ Output	Argument/ Common	Definition
THALT	I	A	Final trajectory time.
IMAN	I	A	Guidance maneuver number.
XØUT	O	A	Output state vector.
MØUT	O	A	Output S/C mass.
THETA	O	A	Output control to state transition matrix, $\tilde{\Theta}_u$.
PHI	O	A	Output state to state transition matrix, Φ.
THRUST	I	C	Array of thrust controls.
BLANK	I	C	Blank common storage of trajectory variables, i.e, the augmented state transition matrix.
TEVNT	I	C	Trajectory event time.
MEVENT	I	C	Trajectory event test flag.
LØCTC	I	C	Location in blank common of the first element in the augmented state transition matrix.
IAUGDC	I	C	Flag used to augment the transition matrix for integration.
TREF	I	C	Initial trajectory time transmitted to TRAJ in seconds.
TDUR	I	C	Final trajectory time transmitted to TRAJ in seconds.
INTEG	I	C	Flag to indicate to TRAJ that the augmented state transition matrix is to be integrated.
ICALL	I	C	TRAJ initialization flag.

Local Variables:

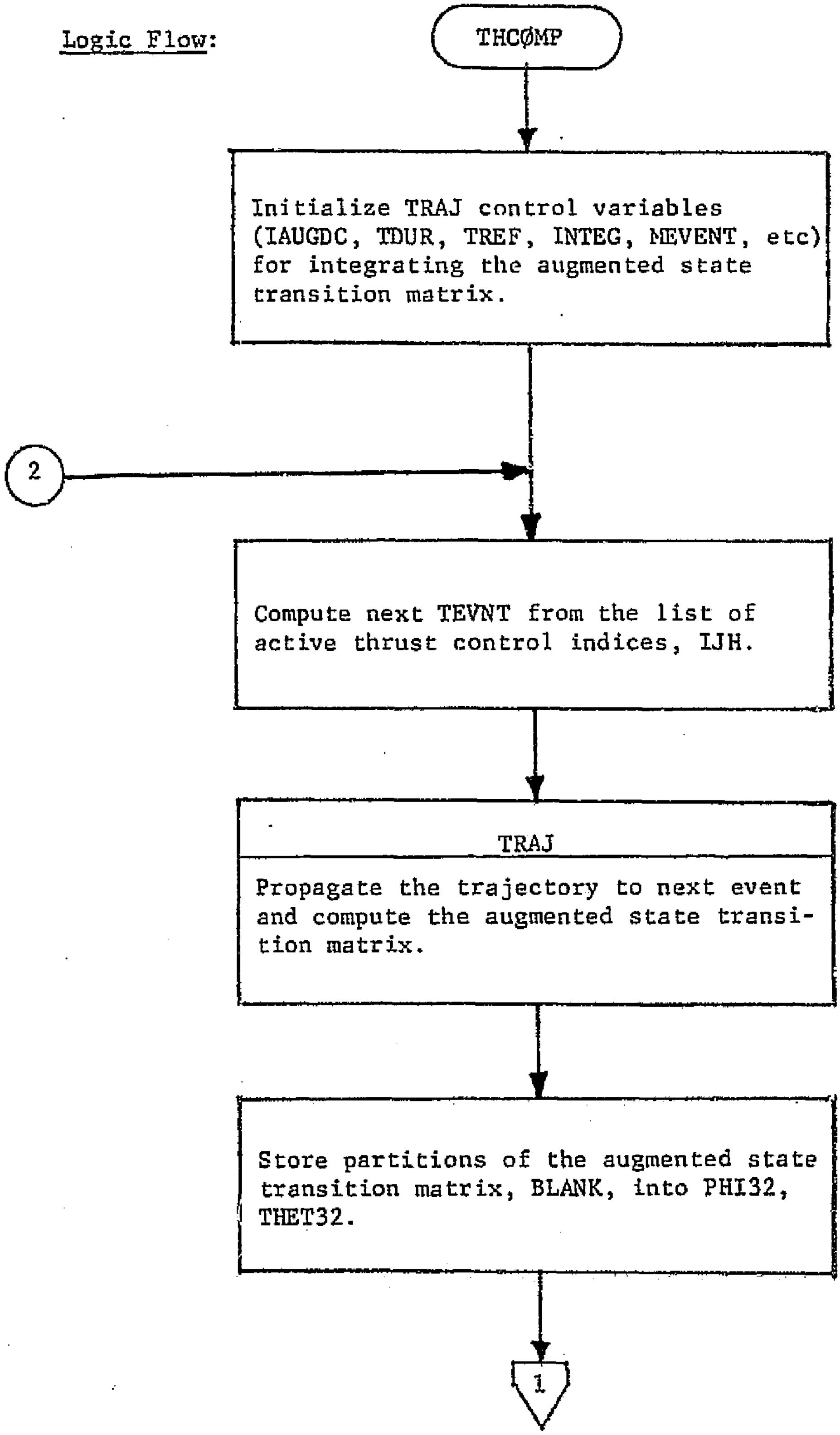
Variable	Definition
NPHI	Dimension of the augmented state transition matrix.
JJ0 JJ1 JJ2	Logic control flag.
PHI21 PHI32	Temporary storage for the Φ matrices output from TRAJ.
THET21 THET32	Temporary storage for the $\hat{\Theta}_u$ matrices output from TRAJ.

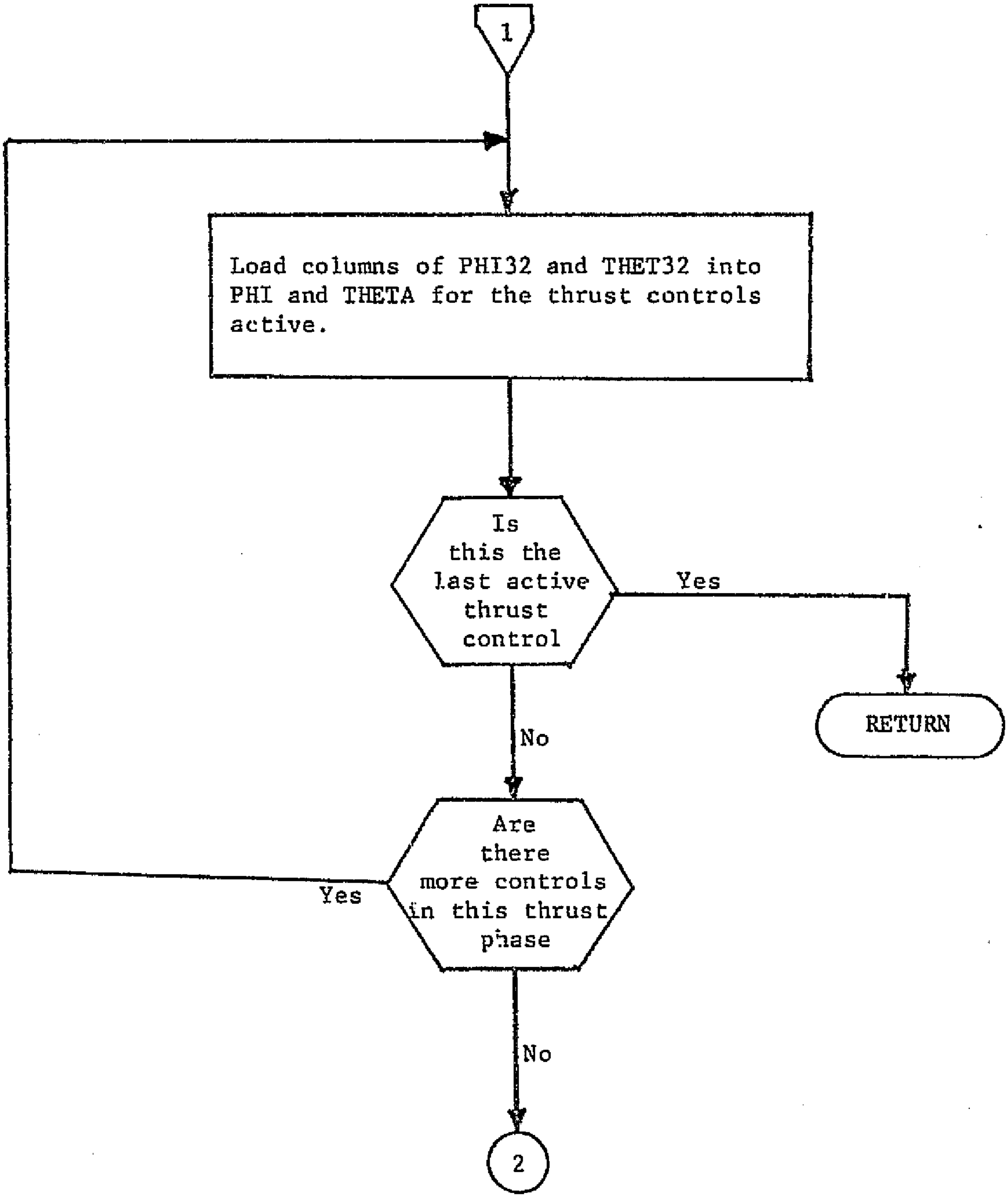
Subroutines Called: CØPY, ICØPY, IDENT, IZERØM, MMAB, MPAK, TRAJ, ZERØM.

Calling Subroutines: STMTAR, REFTRJ, NLGUID.

Common Blocks: CØNST, TIME, TRAJ1, TRAJ2, WØRK, (BLANK).

Logic Flow:


THCØMP
Initialize TRAJ control variables (IAUGDC, TDUR, TREF, INTEG, MEVENT, etc) for integrating the augmented state transition matrix.
2
Compute next TEVNT from the list of active thrust control indices, IJH.
TRAJ
Propagate the trajectory to next event and compute the augmented state transition matrix.
Store partitions of the augmented state transition matrix, BLANK, into PHI32, THET32.
1


1
Load columns of PHI32 and THET32 into PHI and THETA for the thrust controls active.
Is this the last active thrust control
Yes
RETURN
No
Are there more controls in this thrust phase
Yes
No
2

3.7 Subroutine: REFSEP

Purpose: To monitor the subroutine flow in the REFSEP mode of MAPSEP.

Remarks: A complete view of the REFSEP hierarchy is revealed in Section 2.3, page 12-B of this manual.

Subroutines Called: DATREF, TRAK

Calling Subroutines: MAPSEP

Logic Flow: See macrologic listing

3.7.1 Subroutine: DATREF

Purpose: To initialize REFSEP parameters and the trajectory propagator.

Remarks: Proper initialization of the scheduler requires two consecutive calls to subroutine SCHED. Also, TRAJ is called only to in-ialize parameters not to propagate the trajectory.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
GAINCR	O	C	GODSEP variables which are defaulted in DATREF to avoid incorrect computations in subroutine SCHED. None of these variables is relevant to execution of REFSEP.
IGAIN	O	C	
NCNTE	O	C	
NCNTG	O	C	
NCNTP	O	C	
NCNTT	O	C	
NEIGEN	O	C	
NGUID	O	C	
NPRED	O	C	
NTHRST	O	C	
ICALL	O	C	Flag used to initialize TRAJ.
INTEG	O	C	Flag indicating the equations to be integrated in TRAJ.
KARDS	I	C	Number of print schedule cards.

Variable	Input/ Output	Argument/ Common	Definition
LABEL	O	C	Hollerith names of all possible target parameters.
MEVENT	O	C	Flag used to set event detection logic in TRAJ.
MNEXT	O	C	Next scheduled print code.
NSCHED	O	C	Number of print schedule cards.
TCURR	O	C	Current trajectory time.
TEND	I	C	Trajectory end time.
TFINAL	O	C	Trajectory end time.
TM	I	C	Time conversion constant (days to seconds).
TMNEXT	O	C	Time of next print code execution.
TREF	O	C	Initial trajectory time.
TSTART	I	C	Initial trajectory time.

Local Variables: None

Subroutines Called: SCHED, TRAJ

Calling Subroutine: REFSEP

Common Blocks: CØNST, EDIT, LØGIC, MEASI, PRINTH, SCHEDI, SCHEDR, TIME, TRAJ1, TRAJ2, TRKDAT, WØRK

Logic Flow: See listing.

3.7.2 Subroutine: DETAIL (IT)

Purpose: To print trajectory information at the times designated on the formatted schedule cards.

Remarks: The blocks of trajectory infotmation to be printed are cued by the print code which is stored in the variable IT. A discussion of the print code may be found in the User's Manual, Section 2.5, page 52-B.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
APERT	I	C	Gravitational acceleration vectors due to the perturbing bodies.
ATØT	I	C	Total differential acceleration vector.
B	I	C	Magnitude of the B-vector.
BDR	I	C	B · R
BDT	I	C	B · T
BØDY	I	C	Hollerith label of the planets included in the integration.
BV	I	C	Unitary B-vector.
CA	I	C	Closest approach radius computed in BPLANE.
ECC	I	C	Eccentricity.
EPØCH	I	C	Launch epoch
F1	I	C	Hyperbolic anomaly
IPRI	I	C	Flag used to locate information about the primary body.
ISTEP	I	C	Number of integration steps taken.
IT	I	A	Print code.
ITP	I	C	Flag used to locate information about the target body.

Variable	Input/ Output	Argument/ Common	Definition
LØCH	I	C	Blank common location of the step size.
LØCM	I	C	Blank common location of the S/C mass.
LØCYT	I	C	Blank common location of the temporary integrated solution.
MPLAN	I	C	Number of bodies included in the integration.
NPRI	I	C	Planet code of the primary body.
NRECT	I	C	Number of rectifications executed during the trajectory integration.
NTP	I	C	Target planet code.
NTPHAS	I	C	Number of the current control phase.
ØMEGA	I	C	Longitude of the ascending node.
PV	I	C	Unitary peripoint vector.
QV	I	C	Unitary peri-velocity vector.
RAD	I	C	Angular conversion constant (radians to degrees).
SMA	I	C	Semi-major axis.
SØMEGA	I	C	Argument of periapsis.
SV	I	C	Unitary hyperbolic excess velocity vector.
TA	I	C	True anomaly.
TAIM	I	C	Angle between B-vector and T-axis.
TCA	I	C	Time of closest approach computed in BPLANE.
TCURR	I	C	Current event time.
TEVNT	I	C	Current trajectory time.

Variable	Input/ Output	Argument/ Common	Definition
THRACC	I	C	Acceleration vector due to thrust.
TM	I	C	Time conversion constant (days to seconds).
TSI	I	C	Time of SOI crossing as computed in BPLANE.
VENC	I	C	Reference conic position vector.
UP	I	C	Position vectors of all bodies included in the integration.
UREL	I	C	Position vectors of S/C relative to all bodies considered in the integration.
UTRUE	I	C	S/C position vector relative to primary body.
VCA	I	C	Velocity at closest approach as computed in BPLANE.
VENC	I	C	Reference conic velocity vector.
VHP	I	C	Magnitudue of hyperbolic excess velocity.
VP	I	C	Velocity vectors of all bodies considered in the integration.
VREL	I	C	Velocity vectors of S/C relative to all bodies considered in the integration.
VTRUE	I	C	S/C velocity vector relative to the primary body.
WV	I	C	Unitary momentum vector.
XINC	I	C	Ecliptic inclination.
XMEAN	I	C	Mean anomaly.

Local Variables:

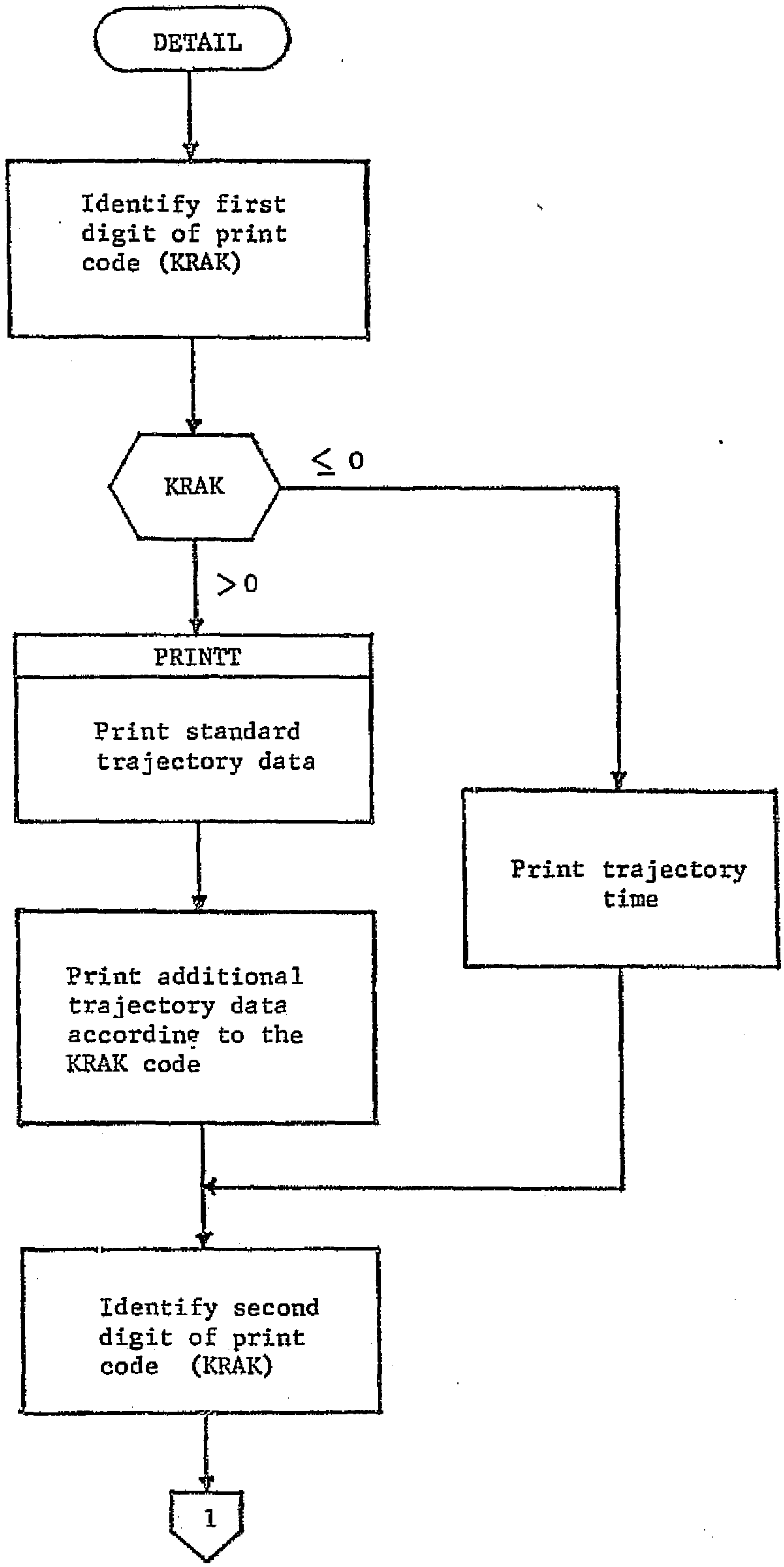
Variable	Definition
ATØTM	Magnitude of total differential acceleration vector.
BVEC	B-vector.
DJ	Julian date of current trajectory time.
IBØD	Primary body code for BPLANE calculations.
KRAK	Intermediate print code.
LBØD	Location of IBØD in the NB array (i.e. IBØD = NB (LBØD)).
PFV	Peri-point vector.
PVV	Peri-velocity vector.
UA	Delta-position vector and delta-velocity vector.
UAM	Magnitude of delta-position vector.
UPM	Heliocentric position magnitudes of bodies considered in the integration.
UR	Unitary position vector of the S/C relative to the primary body.
UV	Unitary velocity vector of the S/C relative to the primary body.
VAM	Magnitude of the delta-velocity vector.
VH	Hyperbolic excess velocity vector.
VPM	Heliocentric velocity magnitude of bodies considered in the integration.

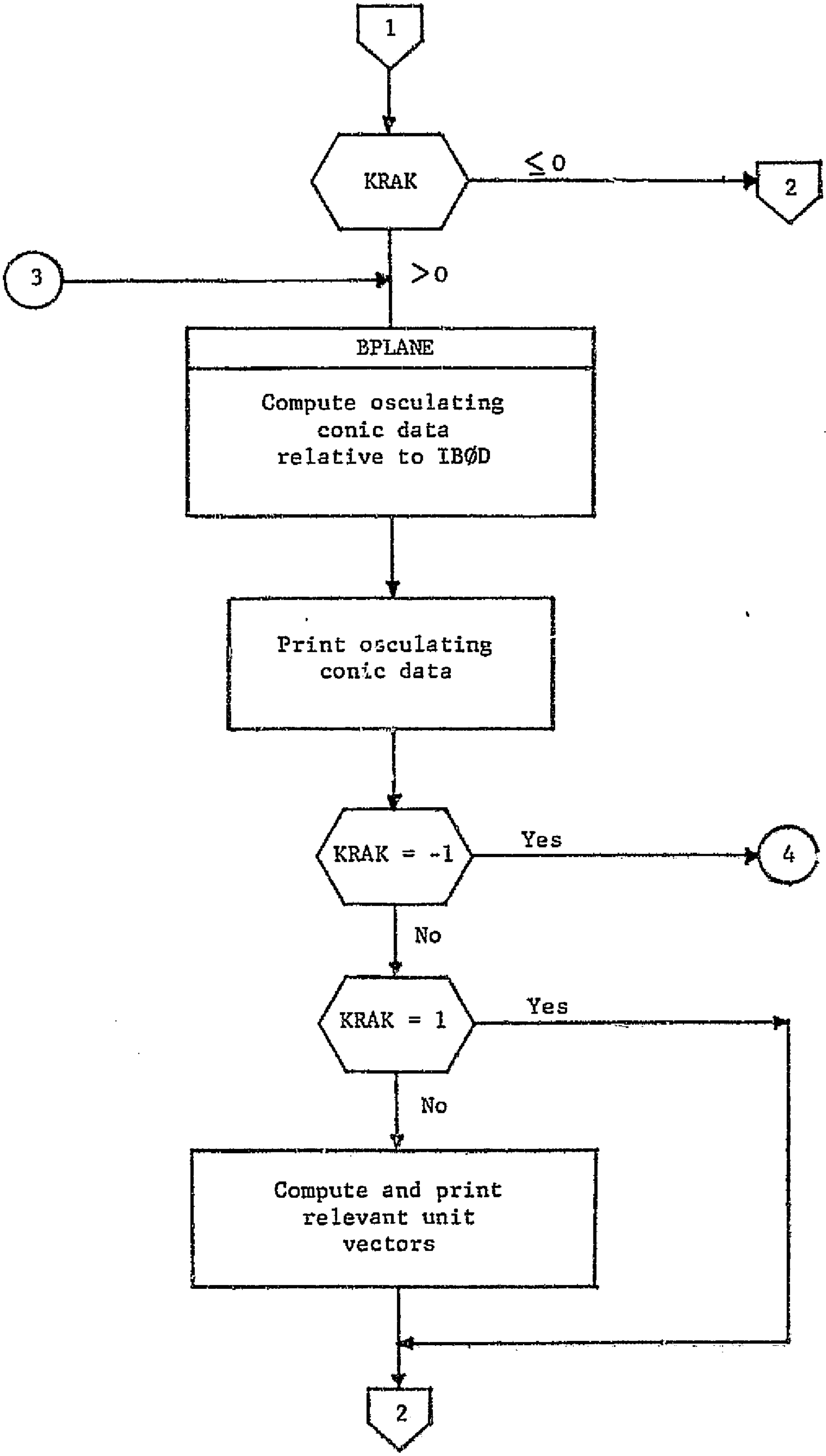
Subroutines Called: BPLANE, CØPY, PRINTI, TSCHED, UDØTV, UNITV, VECMAG

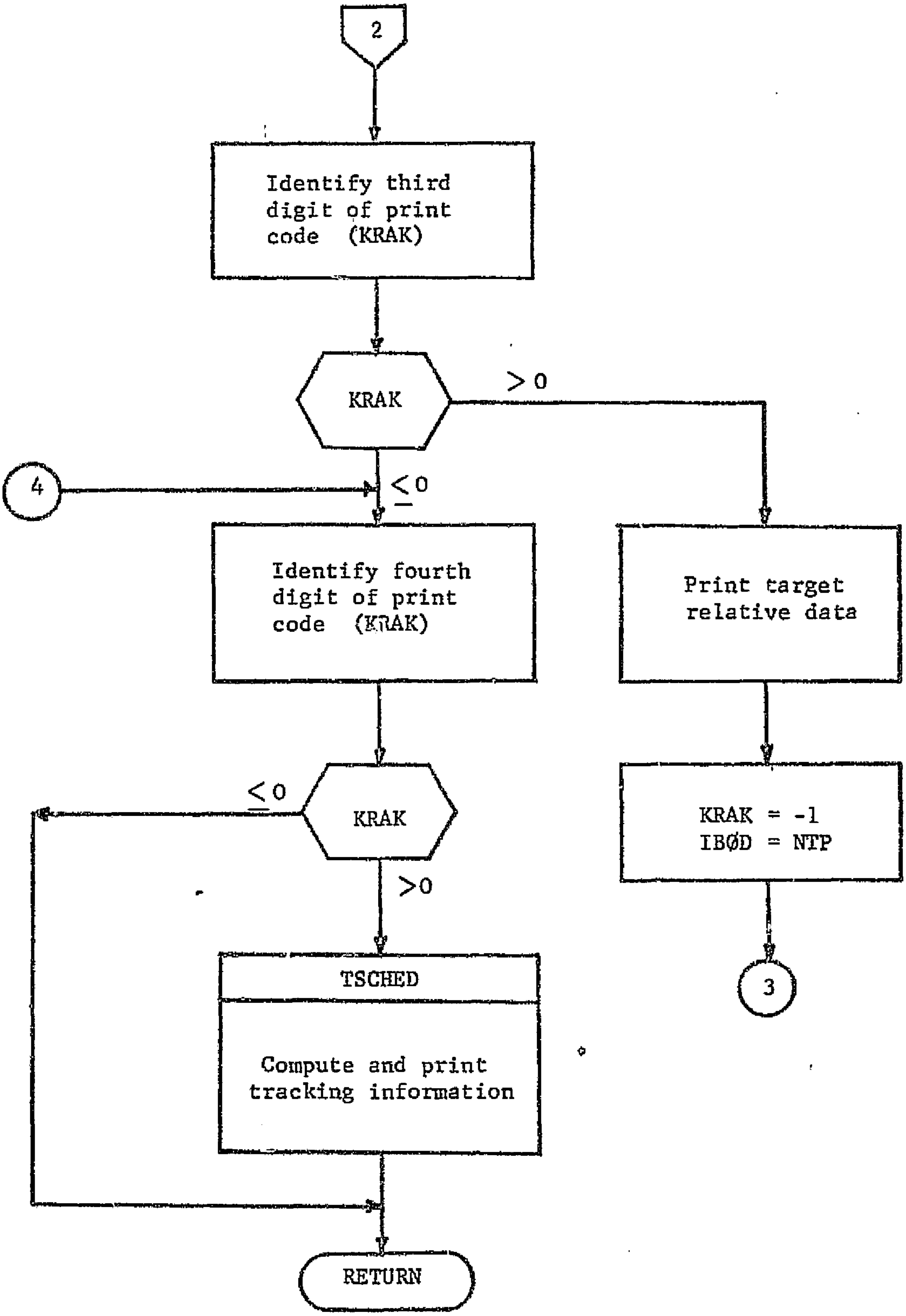
Calling Subroutine: REFSEP

Common Blocks: (BLANK), CØNICS, CØNST, EDIT, SCHEDR TARGET, TIME, TRAJ1, TRAJ2, WØRK

Logic Flow:


DETAIL
Identify first digit of print code (KRAK)
KRAK
≤ 0
> 0
PRINTT
Print standard trajectory data
Print trajectory time
Print additional trajectory data according to the KRAK code
Identify second digit of print code (KRAK)
1


1
KRAK
≤ 0
2
3
> 0
BPLANE
Compute osculating conic data relative to IBØD
Print osculating conic data
KRAK = -1
Yes
4
No
KRAK = 1
Yes
No
Compute and print relevant unit vectors
2


2
Identify third digit of print code (KRAK)
KRAK
> 0
≤ 0
4
Identify fourth digit of print code (KRAK)
Print target relative data
KRAK
≤ 0
> 0
KRAK = -1
IBØD = NTP
3
TSCHED
Compute and print tracking information
RETURN

3.7.3 Subroutine: TRAK

Purpose: To control the point to point (event time to event time) integration of the trajectory propagator.

Remarks: The event times which are input into the trajectory propagator are obtained from the scheduling subroutine SCHED. After TRAJ performs the integration to the desired event time, subroutine DETAIL is called to print detailed trajectory information.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
BDR	O	C	B · R
BDT	O	C	B · T
CA	O	C	Closest approach radius as computed in BPLANE
ECC	O	C	Eccentricity
ISTØP	I	C	Desired trajectory termination flag
ITP	I	C	Target body index (i.e. NTP=NB(ITP))
KUTØFF	O	C	Actual trajectory termination flag
LABEL	I	C	Hollerith labels for terminal conditions
LØCM	I	C	Blank common location of S/C mass
NPRI	I/Ø	C	Primary body code
NTP	I	C	Target body code
ØMEGA	O	C	Longitude of ascending node
RAD	I	C	Angular conversion constant (radians to degrees)
RCA	O	C	Radius of closest approach computed in TRAJ
SMA	O	C	Semi-major axis

Variable	Input/ Output	Argument/ Common	Definition
SØMEGA	O	C	Argument of periapsis
TA	O	C	True Anomaly
TCA	O	C	Time of closest approach computed in BPLANE
TCURR	O	C	Current event time
TEVNT	O	C	Next event time
TM	I	C	Time conversion constant (days to seconds)
TRCA	O	C	Time of closest approach computed in TRAJ
TSI	O	C	Time of SØI crossing computed in BPLANE
TSOI	O	C	Time of SØI crossing computed in TRAJ
TSTART	I	C	Trajectory start time
TSTØP	O	C	Trajectory stop time
UREL	O	C	Position vectors of S/C relative to all bodies considered in the integration
URELM	O	C	Magnitudes of UREL vectors
VCA	O	C	Velocity at closest approach
VHP	O	C	Hyperbolic excess velocity
VREL	O	C	Velocity vector of S/C relative to all bodies considered in the integration
VRELM	O	C	Magnitudes of VREL vectors
XICA	O	C	Inclination of orbit relative to target body
XINC	O	C	Inclination
XMEAN	O	C	Mean anomaly

Local Variables:

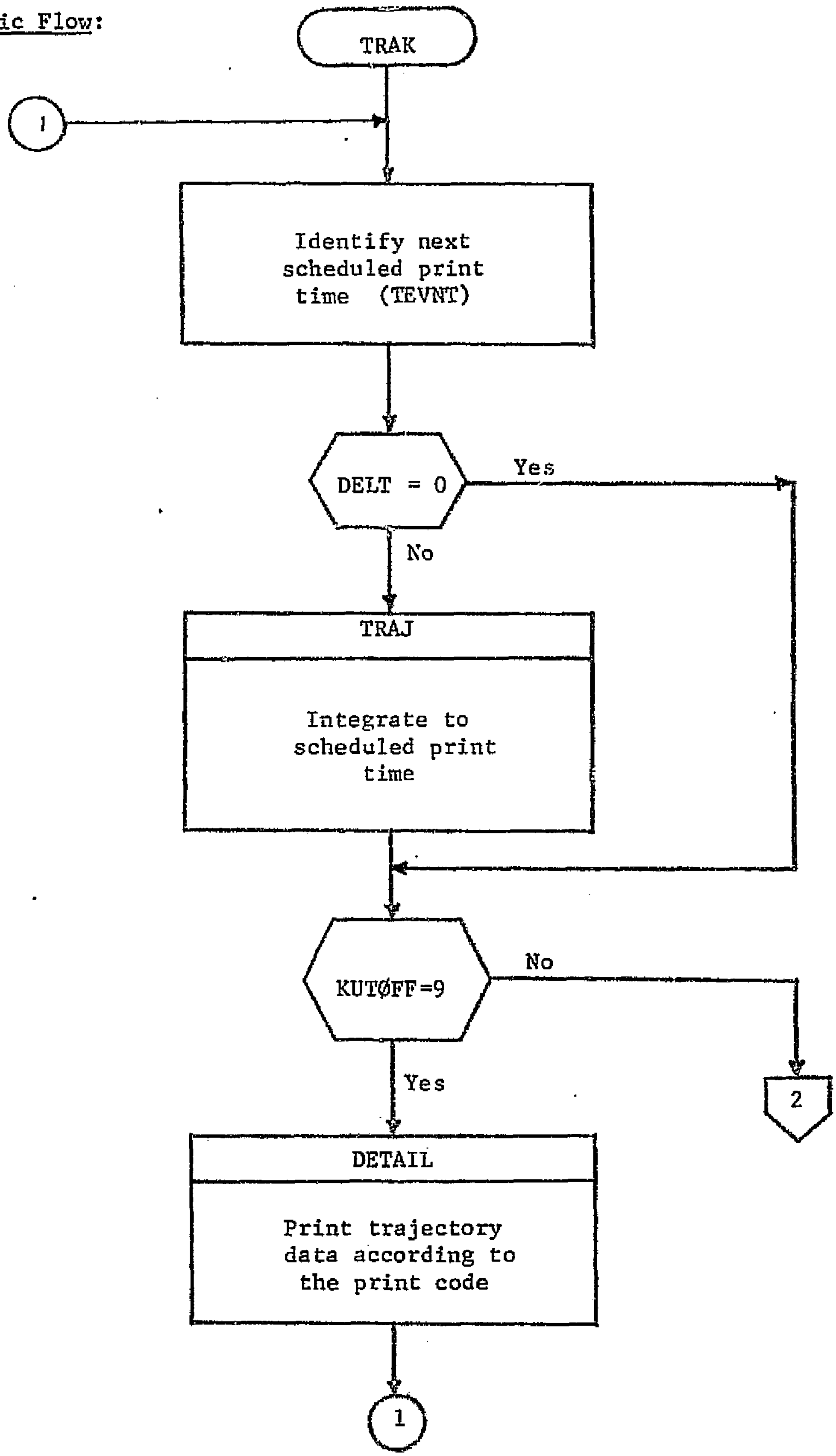
Variable	Definition
DELT	Time between events
ISTØPN	Hollerith labels of requested stopping conditions
JEVNT	Print code
KØFF	Hollerith labels of actual stopping conditions
MISS	Flag indicating whether the target body is the primary body at the trajectory end time

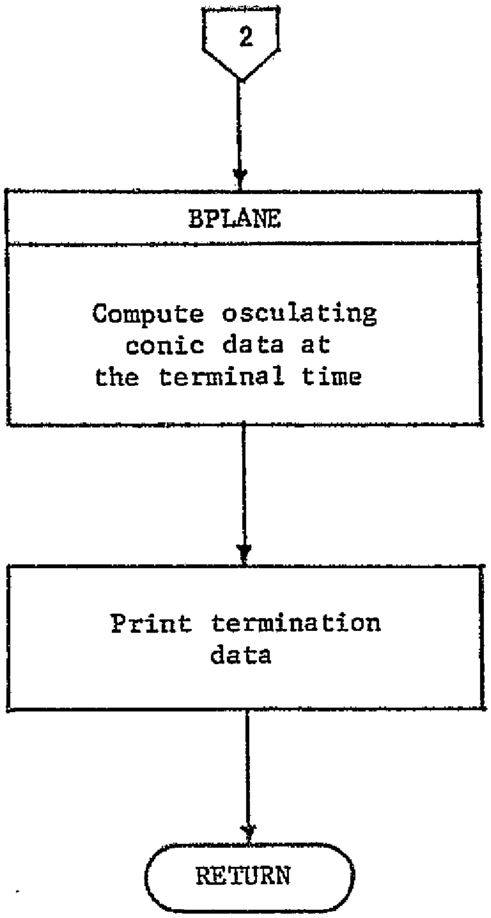
Subroutines Called: BPLANE, DETAIL, SCHED, TRAJ

Calling Subroutine: REFSEP

Common Blocks: (Blank), CØNST, EDIT, EPHEM, PRINTH, SCHEDI, SCHEDR, TARGET, TIME, TRAJ1, TRAJ2, WØRK

Logic Flow:


TRAK
1
Identify next scheduled print time (TEVNT)
DELT = 0
Yes
No
TRAJ
Integrate to scheduled print time
KUTØFF=9
No
Yes
2
DETAIL
Print trajectory data according to the print code
1


2
BPLANE
Compute osculating conic data at the terminal time
Print termination data
RETURN

3.7.4 Subroutine: TSCHED

Purpose: To compute and print S/C tracking information

Method: S/C rise and set times are computed for a selection of tracking stations. The primary assumption, which has been made to simplify the computations, is that the S/C moves very slowly across the celestial sphere. Thus, the rise and set times are poor approximations for near-Earth orbital missions.

Input/Output:

Variable	Input/ Output	Argument/ Common	Definition
ECEQ	I	C	Equatorial to ecliptic transformation matrix
ELVMIN	I	C	Minimum elevation angle
GHZERØ	I	C	Greenwich hour angle at launch
IØBS	I	C	Index of astronomical observatory in STALØC
ITP	I	C	Index of target planet in NB
MPLAN	I	C	Number of bodies considered in the integration
NB	I	C	Vector identifying bodies considered in the integration
NSTA	I	C	Number of S/C tracking stations
NTP	I	C	Target planet code
ØMEGAG	I	C	Earth rotation rate
PI	I	C	π
RAD	I	C	Angular conversion constant (radians to degrees)
STALØC	I	C	Station location coordinates
TCURR	I	C	Current event time

Variable	Input/ Output	Argument/ Common	Definition
TM	I	C	Time conversion constant (days to seconds)
UP	I	C	Heliocentric positions of bodies considered in the integration
UREL	I	C	Position vectors of S/C relative to bodies considered in the integration
URELM	I	C	Magnitudes of UREL vectors
VP	I	C	Heliocentric velocities of bodies considered in the integration
VREL	I	C	Velocity vectors of S/C relative to bodies considered in the integration
VRELM	I	C	Magnitudes of VREL vectors

Local Variables:

Variable	Definition
AZMUTH	Azimuth of S/C relative to the tracking station
DEC	Declination of S/C
ELEV	Elevation of S/C
GECSTA	Geocentric ecliptic station coordinates
GEQSTA	Geocentric equatorial station coordinates
GHA	Greenwich hour angle
GHZERØ	Greenwich hour angle at launch
LAMDA	Right ascension minus Greenwich hour angle
RANGE	S/C range from Earth

Variable	Definition
RHO	S/C range vector
RISE	S/C rise time at each station
RRATE	S/C range rate from Earth
RTA	Right ascension
RVIANG	Range-velocity included angle
SESANG	Sun-Earth-S/C angle
SET	S/C set time at each station
SINELV	sin (ELV)
SLAT	Station latitude
STATE	S/C equatorial state
TM	Time conversion constant (days to seconds)
TWØPI	$2 \times \pi$
UPM	Magnitude of planet position vectors

Subroutines Called: CYEQEC, MMATB, SUB, UDØTV, UNITV, UXV, VECMAG

Calling Subroutine: DETAIL

Common Blocks: CØNST, EDIT, SCHEDR, TIME, TRAJ1, TRAJ2, TRKDAT, WØRK

Logic Flow: See listing

4.0 REFERENCES

1. "MAPSEP, Volume I - Analytical Manual and Volume II - User's Manual," P. Hong, et al, Final Report for NAS8-29666, December, 1973.

2. "Low Thrust Orbit Determination Program - Final Report, NAS1-11686," P. Hong, et al, NASA CR-1'2256, December, 1972.