NASA TECHNICAL NOTE

NASA TN D-3508

c. 1

NASA TN D-3508

A COMPUTATIONAL METHOD FOR TWO-IMPULSE ORBITAL RENDEZVOUS AND TRANSFER PROBLEMS

by Robert L. Collins and Sylvia A. Wallace

Langley Research Center
Langley Station, Hampton, Va.

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION • WASHINGTON, D. C. • AUGUST 1966

ERRATA

NASA Technical Note D-3508

A COMPUTATIONAL METHOD FOR TWO-IMPULSE ORBITAL RENDEZVOUS AND TRANSFER PROBLEMS

By Robert L. Collins and Sylvia A. Wallace

August 1966

Page 64: In the FORTRAN notation in a write statement 14 lines from the bottom of the page, H should be changed to AH. The corrected notation would be as follows:

```
WRITE(6,126)DTTI,DTTF,DTWI,DTWF,RMIN,AH
```

Page 68: Insert the following statements after FORTRAN statement 402 (before 8th line from bottom of page):

```
C           TEST FOR ZERO VALUED DENOMINATOR
            CKDEN=RHO1*COS(TH1)-RHO2*COS(TH2)
            IF(CKDEN.GE.1.0E-10)410,405
      410   CONTINUE
```

Page 70: Insert the following statements after FORTRAN statement 602 (after 7th line from top of page):

```
C           TEST FOR ZERO VALUED DENOMINATOR
            CKDEN=RHO1*COS(TH1)-RHO2*COS(TH2)
            IF(CKDEN.GE.1.0E-10)614,604
      614   CONTINUE
```

Page 72: The absolute value of i_I and i should be used in Subroutine Six which evaluates the transformation matrix. Therefore, the second statement on page 72 should be deleted and replaced by the following two statements:

```
RAII=AII
CALL SIX(A11,A12,A21,A22,A31,A32,OMI,OMEGAI,RAII)
```

Also, the present one-line FORTRAN statement 900 should be deleted and replaced by the following two statements:

```
900  RBI=BI
     CALL SIX(B11,B12,B21,B22,B31,B32,OM,OMEGA,RBI)
```

NASA-Langley, 1968

Issued 9-24-68

Page 78: Add the following statement between lines 1 and 2:

U3=ABS(U3)

Page 83 (table III): Under the heading DV1Z (12th column), delete the minus signs before the first six numbers and insert minus signs before the last nine numbers; under the heading DV2Z (15th column), insert minus signs before all fifteen numbers.

NASA TN D-3508

A COMPUTATIONAL METHOD FOR TWO-IMPULSE ORBITAL RENDEZVOUS AND TRANSFER PROBLEMS

By Robert L. Collins and Sylvia A. Wallace

Langley Research Center
Langley Station, Hampton, Va.

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

For sale by the Clearinghouse for Federal Scientific and Technical Information
Springfield, Virginia 22151 – Price $3.00

A COMPUTATIONAL METHOD FOR TWO-IMPULSE ORBITAL RENDEZVOUS AND TRANSFER PROBLEMS

By Robert L. Collins and Sylvia A. Wallace
Langley Research Center

SUMMARY

A detailed derivation of exact equations and an associated computer program with which the basic parameters involved in transfer or rendezvous between two arbitrary elliptic orbits may be calculated are presented. The computations are exact in the sense that the Kepler solutions for bodies orbiting in a point-gravity field are used. This method deviates from the "classical" approach based on the theorem of Lambert, inasmuch as it uses the true anomaly and the Kepler equations for iterating to the desired rendezvous transfer time. The method has a unique feature in that definite boundaries, dependent on the problem input, are used which limit the range of the true anomaly, and thereby reduce the search effort required in the iteration procedure. The program provides for a solution to problems where the transfer angle is less than 360°. Examples are given for three particular uses of the program: (1) interplanetary transfer between massless planets, (2) near-planet orbit rendezvous, and (3) orbital transfer.

INTRODUCTION

It seems appropriate to provide a useful technique for the computation of velocity increments and other important parameters involved in the problems of orbital rendezvous and transfer by use of the solution to the exact equations of motion. Various programs and techniques exist at this time; however, they are either unpublished or are not of a sufficiently general nature to be used in the variety of orbital problems one might desire. (For instance, see refs. 1 to 4.) It is desirable, therefore, to have a simple and yet general computational method which will solve the problems of: (a) interplanetary transfer between massless planets, (b) planet orbit rendezvous, and (c) orbital transfer. The purpose of this paper is to give the description of the analysis required for the solution of the rendezvous problem from the exact Keplerian relations. The fundamental problem is that of determining the velocity increment required to rendezvous from some initial interceptor orbit to some final target orbit as a function of transfer time. Also determined are eccentricity, semi-major axis, initial and final anomalies, and other parameters associated with the transfer orbit. Although the problem, as stated, is a rendezvous problem, it is also possible to interpret the results for use in studying the orbital transfer problem. It is assumed that the Keplerian orbital quantities are known in advance for the target vehicle and that the initial interceptor orbit is known either from its Keplerian orbital elements or from relative coordinate data. The coordinates chosen are referenced to the target orbit, and another axis transformation will be necessary if the user desires the results and input referenced to some other axis system (such as the ecliptic). Figures 1, 2, and 3 show the coordinates and position symbols used to describe the orbits and figures 4 and 5 show the (input) information needed on the position of the orbits.

A solution to the rendezvous problem may be obtained by specifying the transfer time for the interceptor to travel from its initial orbit to its final (target) orbit which, along with the initial conditions of the problem, gives the initial and final positions in space through which the interceptor must pass. The transfer orbit must then be found which passes through the known initial and final positions. An iteration is required for this calculation inasmuch as the transfer orbit which will yield the proper transfer time is not yet known. The procedure begins by choosing some orbit which passes through the initial and final position vectors and then the corresponding transfer time is computed and compared with the real (desired) transfer time. If the computed transfer time is not the same as the desired time, another orbit must be chosen and the time again computed and compared. This process is repeated until the conic section which provides the true rendezvous transfer orbit is found.

There are (at least) two methods which have been used for this iteration. The solution by use of Lambert's theorem was used by Battin (ref. 1) and Breakwell (ref. 2). However, iteration of the true anomaly, as was considered by Lascody (ref. 3), is more readily visualized since it does not require such artificial devices as "flattening" the transfer orbit used in geometrically describing the transfer problem deduced from the theorem of Lambert (ref. 1). It is also possible to find definite regions within which the true anomaly must be chosen and therefore shorten computational (convergence) time to the extent that a direct search routine may be used in the iteration; this routine does not involve the derivatives necessary in the method of iteration as presented in reference 3.

In programing this problem, an effort has been made to provide a sufficiently general program with a minimum amount of difficulty in reading input and printing output information for the particular problem. For instance, problem input information may be provided in two fundamentally different ways: (1) as Keplerian data (eccentricity, semimajor axis, longitude of ascending node, etc.) and (2) as relative coordinate input referenced to rectangular coordinate axes fixed with the target vehicle.

SYMBOLS

Unless otherwise noted all quantities are nondimensional. The nondimensional forms are derived as follows:

$$\text{Nondimensional length} = \frac{\text{Dimensional length}}{\text{Semi-major axis of target orbit, } a_T}$$

$$\text{Nondimensional velocity} = \frac{\text{Dimensional velocity}}{\text{Mean circular velocity of target, } V_{CT}}$$

$$\text{Nondimensional angular rate} = (\text{Dimensional angular rate}) \times \frac{\text{Target orbital period, } P_T}{2\pi}$$

$$\text{Nondimensional time} = \frac{\text{Dimensional time}}{\text{Target orbital period, } P_T}$$

a	semi-major axis of orbit
a_{ij}	element of transformation matrix, x,y,z to X,Y,Z
b_{ij}	element of transformation matrix, x',y',z' to X,Y,Z
c	chord joining ρ_1 to ρ_2
e	eccentricity of orbit
E	eccentric anomaly of vehicle in orbit
F	functional relationship
F_1,F_2	equivalent eccentric anomalies for hyperbolic orbit
h	increment for iteration
H	angular momentum
i	inclination of orbital plane to target orbit plane
$\hat{i},\hat{j},\hat{k}$	unit vectors along x,y,z; x'',y'',z'' axes as indicated
$\hat{i}',\hat{j}',\hat{k}'$	unit vectors along x',y',z' axes
$\hat{I},\hat{J},\hat{K}$	unit vectors along X,Y,Z axes

M	mean anomaly
p	semi-latus rectum
P_T	target orbital period, $2\pi\sqrt{\frac{a_T^3}{\mu}}$, seconds
r	radial distance to target
$\dot{r}_f(X), \dot{r}_f(Y), \dot{r}_f(Z)$	velocity components after final impulse
t	time
t_o	time at observation of input
t_i	time of initial impulse
t_f	time of final impulse
Δt_w	wait time before initial impulse, $t_i - t_o$
Δt	interceptor transfer time, $t_f - t_i$
T_1, T_2	computed times from periapse when the interceptor is at ρ_1 and ρ_2 in transfer orbit
ΔT	computed transfer time, $T_2 - T_1$, for comparison
V	speed
V_{CT}	mean circular speed of target orbit, $\sqrt{\frac{\mu}{a_T}}$, ft/sec
$V(X), V(Y), V(Z)$	velocity components along X,Y,Z coordinate axes
$\Delta V(X), \Delta V(Y), \Delta V(Z)$	velocity increment components
$\Delta V_i, \Delta V_f$	initial velocity impulse; final velocity impulse
ΔV	total velocity impulse
x, y, z	coordinates fixed to interceptor initial orbit (inertial)
x', y', z'	coordinates fixed to interceptor transfer orbit (inertial)
x'', y'', z''	coordinates fixed to target vehicle (rotating)

X,Y,Z	coordinates fixed to target orbit (inertial)
$\alpha, \bar{\alpha}, \beta$	auxiliary quantities
γ	constant determining hyperbolic or elliptic computation
$\Delta\theta$	transfer angle
θ	true anomaly of interceptor in transfer orbit
$\delta\theta$	increment of θ_1
δt	increment in transfer time
δt_w	increment in wait time
$\delta\nu$	increment in true anomaly of interceptor in initial orbit
μ	gravitational constant, ft^3/sec^2
ν	true anomaly of interceptor in initial orbit
ξ	radial distance to interceptor in initial orbit
$\dot{\xi}_i(x), \dot{\xi}_i(y), \dot{\xi}_i(z)$	velocity components of interceptor immediately before initial impulse
ρ	radial distance to interceptor in target orbit
$\dot{\rho}(x'), \dot{\rho}(y'), \dot{\rho}(z')$	velocity components of interceptor in transfer orbit
ϕ	true anomaly of target
ψ	auxiliary angle
ω	longitude of periapsis measured from node in plane in question
Ω	longitude of ascending node measured in target plane

Subscripts:

c	circular orbits
f	final time $t = t_f$
i	initial time $t = t_i$ unless specified differently
I	initial interceptor orbit

k	arbitrary index
o	observation time $t = t_o$
p	parabolic orbit
T	target orbit
w	wait time
1,2	used to distinguish between the two terminal vectors in transfer orbit iterations
1,2,3	used to indicate vector components along indicated Cartesian coordinates
1,2,3,4	indicates velocity components immediately before and after initial (1,2) and final (3,4) impulses
X,Y,Z	used to indicate vector components along X,Y,Z coordinate axes

Nonsubscripted Keplerian orbital parameters refer to the transfer orbit. Dots over symbols denote derivatives with respect to time; a caret (^) over a symbol denotes vector quantities.

DISCUSSION OF ANALYSIS

The coordinate systems are chosen with the target orbit plane as the reference plane, the periapsis of the target defining the reference position vector in space. Figure 1 is presented to emphasize the parameters involved in describing the target orbit and the target vehicle. The target vehicle radius is denoted by r and its true anomaly by ϕ. Input is given to define the geometry of the target orbit e_T, a_T and the true anomaly of the target vehicle ϕ_o at time $t = t_o$. The coordinate system X, Y, and Z is defined with X piercing the target orbit periapse, Z pointing along the positive rotation vector of the target, and Y completing a right-handed triad.

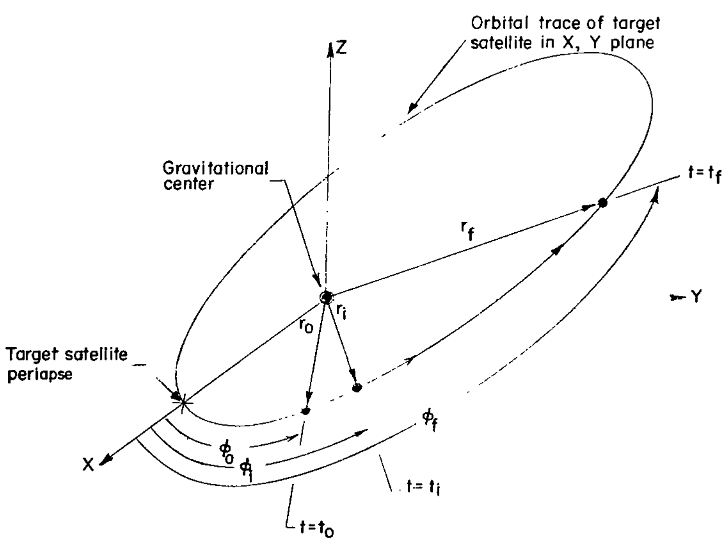


Figure 1.- Target satellite orbit and associated quantities.

Unless otherwise noted, the quantities in this study are all nondimensional for generality. This condition allows the deletion of two parameters μ, and a_T from the pertinent equations. The semi-major axis a_T is not needed when dimensionless quantities are considered inasmuch as it is used for the normalizing and is only necessary when certain dimensional information is required. Neither is the central-body gravitational constant μ required for dimensionless studies but must be used if the times are required to be dimensional (minutes, days).

The parameters required to describe the initial interceptor plane are shown in figure 2. These parameters are the usual Keplerian elements describing the initial plane and the periapse position of the interceptor orbit Ω_I, ω_I, i_I and the geometrical elements of the interceptor ellipse e_I, a_I. The position vector of the interceptor in the initial interceptor orbit is given by the magnitude ξ and the true anomaly ν. The coordinates x, y, and z are also defined in this system; x being directed through the periapse, z perpendicular to the orbital plane (see fig. 2), and y completing a right-handed triad.

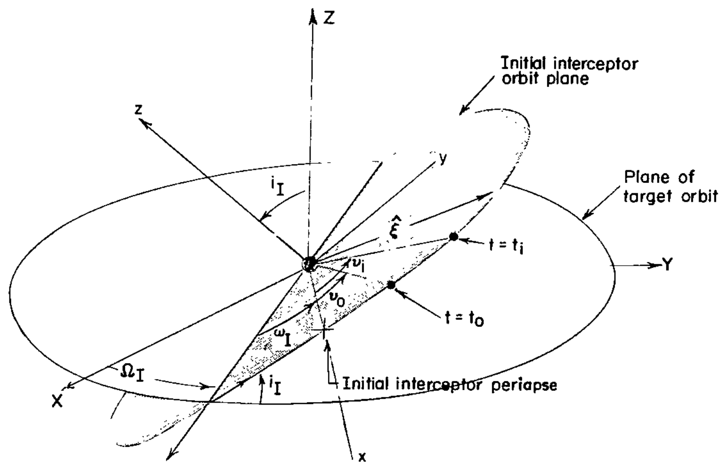


Figure 2.- Plane of initial interceptor orbit and associated quantities.

It is not necessary to specify the six parameters for the interceptor position in the Keplerian form $\left(\Omega_I,\ \omega_I,\ i_I,\ e_I,\ a_I, \text{ and } \nu_0\right)$ as the program provides an alternate form for this input. The position and velocity $\left(x_0'',\ y_0'',\ z_0'',\ \dot{x}_0'',\ \dot{y}_0'', \text{ and } \dot{z}_0''\right)$ of the interceptor relative to the target at time $t = t_0$ may be used in place of the Keplerian parameters. (See fig. 5.) A derivation of the relationship between these parameters is given in appendix A.

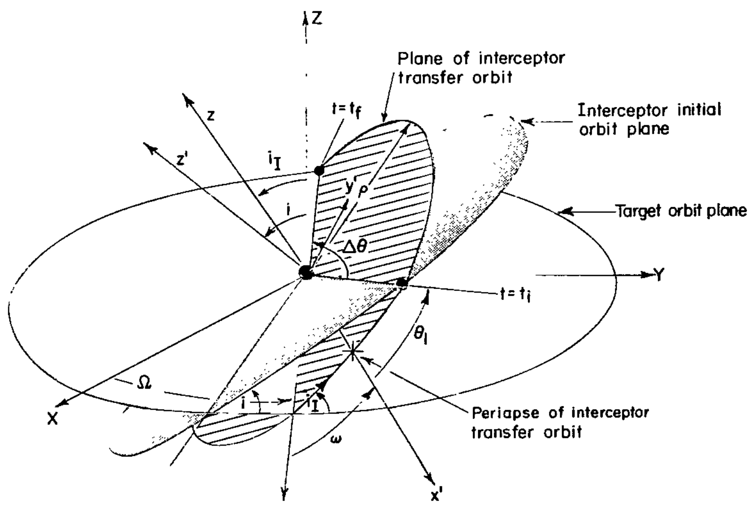


Figure 3.- Interceptor transfer orbit and associated quantities.

The transfer orbit plane is specified entirely by the geometry of the initial and final position vectors of the interceptor.

(See fig. 3.) The time for the initial impulse is denoted by $t = t_i$ and the final impulse $t = t_f$. The quantities ϕ_o and ν_o are the true anomalies of the target and interceptor at some time $t = t_o$ but the initial impulse may not come until some time later t_i; thus, another quantity is introduced, the wait time Δt_w where $\Delta t_w = t_i - t_o$. It is then clear that the transfer time Δt is $\Delta t = t_f - t_i$. The initial position vector $\hat{\xi}_i$ is determined by the orbital quantities and the wait time e_I, a_I, Ω_I, i_I, ω_I, ν_o, and Δt_w or the relative coordinate data x''_o, y''_o, z''_o, $\dot{x}''_o$, $\dot{y}''_o$, $\dot{z}''_o$, and Δt_w. To determine the final position vector $\hat{r}_f$, the position vector of the target at $t = t_f$, the quantities e_T, ϕ_o, and $\Delta t_w + \Delta t$ are needed. The total time from t_o to t_f is the sum of the wait time and the transfer time.

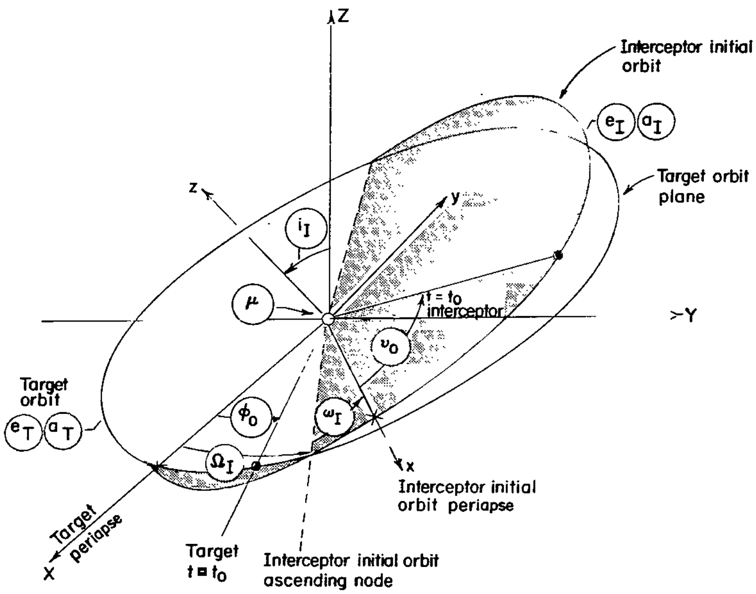


Figure 4.- Quantities required for Keplerian input to program (DATAS = 1.0).

The initial position vector $\hat{\xi}_i$ is determined by first computing the eccentric anomaly E_{Io} at $t = t_o$ of the interceptor in its initial orbit from the equation

$$\tan \frac{E_{Io}}{2} = \sqrt{\frac{1 - e_I}{1 + e_I}} \tan \frac{\nu_o}{2} \quad (1)$$

The mean anomaly at $t = t_o$ is found directly from Kepler's equation as

$$M_{Io} = E_{Io} - e_I \sin E_{Io} \quad (2)$$

and at $t = t_i$ the mean anomaly will be M_{Ii} where

$$M_{Ii} = M_{Io} + 2\pi\sqrt{\frac{1}{a_I^3}}\, \Delta t_w \quad (3)$$

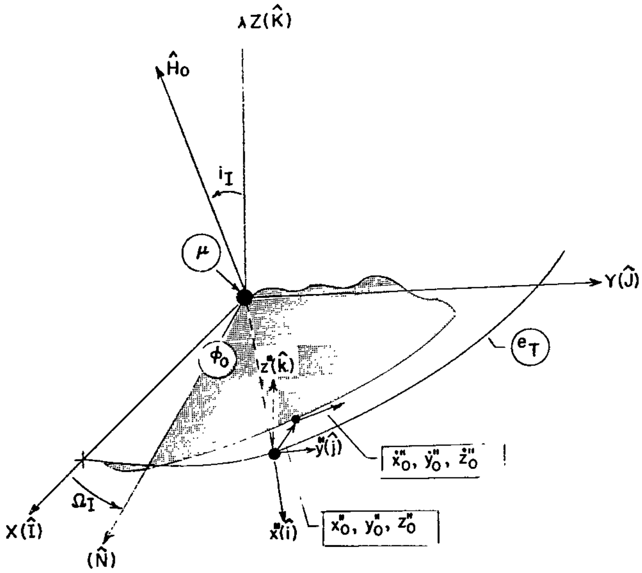


Figure 5.- Quantities required for relative coordinate input to program (DATAS = 2.0).

The eccentric anomaly E_{Ii} at $t = t_i$ is found from iteration of the Kepler equation as described in reference 5.

$$M_{Ii} = E_{Ii} + e_I \sin E_{Ii} \tag{4}$$

After obtaining E_{Ii} from equation (4), the true anomaly ν_i may be found from

$$\tan \frac{\nu_i}{2} = \sqrt{\frac{1 + e}{1 - e}} \tan \frac{E_{Ii}}{2}$$

After the true anomaly of the interceptor ν_i preceding the first impulse is found, the radius vector magnitude is obtained from the well-known solution

$$\xi_i = \frac{a_I\left(1 - e_I^2\right)}{1 + e_I \cos \nu_i} \tag{5}$$

The rates may be easily computed from the relations

$$\dot{\nu}_i = \frac{\sqrt{a_I\left(1 - e_I^2\right)}}{\left(\xi_i\right)^2} \tag{6}$$

and

$$\dot{\xi}_i = \frac{e_I \sin \nu_i}{\sqrt{a_I\left(1 - e_I^2\right)}} \tag{7}$$

The initial vector $\hat{\xi}_i$ written in the x,y,z coordinate system (fig. 2) becomes

$$\hat{\xi}_i = \xi_I\left(\cos \nu_i \hat{i} + \sin \nu_i \hat{j}\right) \tag{8}$$

The final position vector is determined from the initial conditions e_T and ϕ_o along with the elapsed time $\Delta t_w + \Delta t$ in the same manner as the initial position vector. These computations are shown in appendix B. By repeating these steps for the target vehicle at $t = t_f$ the true anomaly ϕ_f may be obtained and may be used to find the final radius vector written in the X,Y,Z system as

$$\hat{r}_f = r_f\left(\cos \phi_f \hat{I} + \sin \phi_f \hat{J}\right) \tag{9}$$

where

$$r_f = \frac{1 - e_T^2}{1 + e_T \cos \phi_f} \tag{10}$$

Also

$$\dot{\phi}_f = \frac{\sqrt{1 - e_T^2}}{\left(r_f\right)^2} \tag{11}$$

$$\dot{r}_f = \frac{e_T \sin \phi_f}{\sqrt{1 - e_T^2}} \tag{12}$$

The interceptor velocity immediately preceding the initial impulse at $t = t_i$ is obtained from

$$\dot{\hat{\xi}}_i = \left(\dot{\xi}_i \cos \nu_i - \xi_i \dot{\nu}_i \sin \nu_i\right)\hat{i} + \left(\dot{\xi}_i \sin \nu_i + \xi_i \dot{\nu}_i \cos \nu_i\right)\hat{j} \tag{13}$$

and after the final impulse the interceptor velocity must be the same as that of the target or

$$\dot{\hat{r}}_f = \left(\dot{r}_f \cos \phi_f - r_f \dot{\phi}_f \sin \phi_f\right)\hat{I} + \left(\dot{r}_f \sin \phi_f + r_f \dot{\phi}_f \cos \phi_f\right)\hat{J} \tag{14}$$

The transfer orbital plane and some of the properties of the transfer orbit are found by noting that the interceptor must leave vector $\hat{\xi}_i$ at time t_i and arrive at vector $\hat{r}_f$ at time t_f. Since $\hat{\xi}_i$ and $\hat{r}_f$ are known, the transfer plane properties and the transfer angle $\Delta\theta$ may be determined.

In equations (8) and (9), $\hat{\xi}_i$ was specified in terms of its components in the x,y,z coordinate system whereas $\hat{r}_f$ was specified in the X,Y,Z system. In order to manipulate with these vectors, their components must be referenced to the same coordinate system. The target coordinate system X,Y,Z is used here so that the vector $\hat{\xi}_i$ must be transformed. This transformation can be accomplished by the common Euler angle matrix. The elements of this matrix are dependent upon the angles Ω_I, ω_I, and i_I and are given in appendix B. The matrix will here be simply noted as $\left[a_{ij}\right]$. Then $\hat{\xi}_i$ is written with components in X,Y,Z as

$$\hat{\xi}_i\left(\hat{I},\hat{J},\hat{K}\right) = \left[a_{ij}\right]\xi_i\left(\hat{i},\hat{j},\hat{k}\right) \tag{15}$$

or letting

$$\hat{\xi}_i\left(\hat{I},\hat{J},\hat{K}\right) = \xi_i(X)\hat{I} + \xi_i(Y)\hat{J} + \xi_i(Z)\hat{K} \tag{16}$$

where $\xi_i(X)$, $\xi_i(Y)$, $\xi_i(Z)$ are functions of ξ_i, ν_i, ω_i, Ω_i, and i_I and are found from equation (15). These relations are written out in appendix B (eqs. (B21)).

With both $\hat{\xi}_i$ and $\hat{r}_f$ in the X,Y,Z coordinate system, the transfer angle $\Delta\theta$ and the transfer-orbit plane inclination i may be derived from the vector identities

$$\hat{\xi}_i \cdot \hat{r}_f = \xi_i r_f \cos \Delta\theta$$

$$\hat{\xi}_i \times \hat{r}_f = \xi_i r_f \sin \Delta\theta \hat{k}'$$

where

$$\hat{k}' = -\sin i \sin \phi_f \hat{I} + \sin i \cos \phi_f \hat{J} + \cos i \hat{K}$$

A complete derivation of $\Delta\theta$ and i is given in appendix B.

The velocity components of equation (13) may also be transformed to the X,Y,Z system

$$\dot{\hat{\xi}}_i(\hat{I},\hat{J},\hat{K}) = \left[a_{ij}\right]\dot{\hat{\xi}}(\hat{i},\hat{j},\hat{k}) \tag{17}$$

which gives expressions for $\dot{\xi}_i(X)$, $\dot{\xi}_i(Y)$, and $\dot{\xi}_i(Z)$ as shown in appendix B (eqs. (B44)).

The problem now becomes one of finding an arc of a conic section which will pass through the initial vector $\hat{\xi}_i$ and the final vector $\hat{r}_f$ and which also has the property that a body traversing this arc will do so in the desired transfer time Δt. For future convenience, the following convention is defined: Let ρ_1 be the minimum of r_f and ξ_i and ρ_2 the maximum; then the transfer orbit must be the conic section passing through the radii ρ_1 and ρ_2, separated by an angle $\Delta\theta$, in the time Δt.

If an angle θ_1 of periapse to ρ_1 is guessed, the eccentricity for the conic section which has radii ρ_1 and ρ_2 separated by $\Delta\theta$ may be found from

$$\left.\begin{aligned} \rho_1 &= \frac{p}{1 + e \cos \theta_1} \\ \rho_2 &= \frac{p}{1 + e \cos \theta_2} \end{aligned}\right\} \tag{18a}$$

where

$$\theta_2 = \theta_1 + \Delta\theta \tag{18b}$$

and

$$p = a\left(1 - e^2\right) \tag{18c}$$

from which p is eliminated to find

$$e = \frac{\rho_2 - \rho_1}{\rho_1 \cos \theta_1 - \rho_2 \cos \theta_2} \tag{19}$$

The eccentric anomalies may also be computed from

$$\tan \frac{E}{2} = \sqrt{\frac{1 - e}{1 + e}} \tan \frac{\theta}{2}$$

and the mean anomalies from

$$M = E - e \sin E$$

The time of transfer for this particular choice of θ_1 is found from the mean anomalies as

$$\Delta T = \frac{\sqrt{a^3}}{2\pi} \Delta M$$

The semi-major axis a is determined from e and θ_1 or θ_2 in equation (18c).

The computed transfer time ΔT will not, in general, correspond to the desired transfer time Δt; therefore, it will be necessary to assume a new value for θ_1 and continue with this process until the desired agreement is obtained.

It is possible to find certain regions from which to choose θ_1 and thereby shorten the iteration considerably. It is easy to see in which conic section the transfer orbit must be by computing the time required to transfer by a parabolic orbit. This time may be determined without iteration by setting $e = 1.0$ and following the procedure outlined in appendix B. If this parabolic time ΔT_p is less than the desired transfer time Δt, the orbit must be elliptic ($e < 1.0$) whereas if the parabolic time ΔT_p is greater than the time Δt, the orbit must be hyperbolic ($e > 1.0$). The procedure is to find the type of orbit, hyperbolic or elliptic, and then set θ_1 equal to the parabolic anomaly θ_p plus or minus some increment $\delta\theta_1$ so that θ_1 lies in the hyperbolic or elliptic region, whichever is appropriate. The size of this increment $\delta\theta_1$ is important in convergence of the iteration. It is governed by two other limiting values, one each for the elliptic and hyperbolic regions. Then a straightforward incrementation process is used directly until the correct value of θ_1 is found. A complete description of this process is found in appendix B.

After the properties e, a, θ_1, $\Delta\theta$, p, Ω, ω, and i of the transfer orbit are found, the next step is to compute the velocity increments required for the rendezvous. The velocities before the initial impulse and after the final impulse are given in equations (17) and (14). The velocities after the initial impulse and before the final impulse are described in the transfer orbit. If $\hat{i}'$, $\hat{j}'$, and $\hat{k}'$ are the unit vectors along the x',y',z' coordinate axes previously defined in the transfer orbit system, these velocities are written as

$$\dot{\hat{\rho}}_i = \left(\dot{\rho}_i \cos \theta_i - \rho_i\dot{\theta}_i \sin \theta_i\right)\hat{i}' + \left(\dot{\rho}_i \sin \theta_i + \rho_i\dot{\theta}_i \cos \theta_i\right)\hat{j}'$$

$$\dot{\hat{\rho}}_f = \left(\dot{\rho}_f \cos \theta_f - \rho\,\dot{\theta}_f \sin \theta_f\right)\hat{i}' + \left(\dot{\rho}_f \sin \theta_f + \rho_f\dot{\theta}_f \cos \theta_f\right)\hat{j}'$$

These velocities are then transformed to the X,Y,Z system for ease of manipulation. This transformation is done by a matrix $\left[b_{ij}\right]$ identical functionally to $\left[a_{ij}\right]$ but with the transfer orbit angular parameters Ω, ω, and i replacing the parameters for the interceptor initial orbit Ω_I, ω_I, and i_I. After these transformations the velocity increments are easily found from

$$\Delta\hat{V}_i = \dot{\hat{\rho}}_i - \dot{\hat{\xi}}_i$$

$$\Delta\hat{V}_f = \dot{\hat{r}}_f - \dot{\hat{\rho}}_f$$

and the total velocity increment required becomes

$$\Delta V = \sqrt{\left|\Delta\hat{V}_i\right|^2 + \left|\Delta\hat{V}_f\right|^2}$$

A thorough discussion of this method and the pertinent mathematics of the problem is found in appendix B.

COMPUTER INPUT AND OUTPUT

The problem was programed in the FORTRAN IV language for the IBM 7094 computer installation at Langley Research Center. Certain options were incorporated in the input of the program and are described. A more thorough description of the entire program is found in appendix C. It was felt that the present writeup is sufficient for most uses and if changes are desired, they are easily incorporated through the FORTRAN IV language.

The program automatically increments the anomaly ν_O of the interceptor at $t = t_O$, the wait time in orbit Δt_W, and the transfer time Δt. The incrementation begins with Δt or Δt_W depending on the value of the option control variable GUIDE in the input. The second parameter incremented is Δt_W or Δt, whichever is not incremented first, and the third incrementation is ν_O.

An option is also available for using the input and output times Δt and Δt_W as dimensional quantities. The input quantity OPTION determines whether the quantities Δt and Δt_W are dimensionless or dimensioned as minutes or as days. Because of the nondimensional parameters used in the program, the input μ and a_T may be any convenient values as long as the time is dimensionless and P_T and V_{CT} are not desired; otherwise, μ and a_T must be properly dimensioned.

The input quantity DATAS determines whether the initial interceptor orbit data are Keplerian or relative Cartesian.

Other necessary input quantities are h, the increment constant for the θ_1 iteration (in the examples solved a value of 3 was used), δt and δt_W, the transfer and wait time incrementation for the time grid. Also, input is the quantity r_{min} which is the minimum radius the interceptor may take as it travels the transfer arc. This value does not affect the computations in any way; however, if the radial distance of the transfer arc falls below the value r_{min}, an asterisk is printed out at the right-hand side of the output sheet.

The variables output in the program described herein were those which were considered to be of general use. The output is printed as shown in tables I, II, and III which are output data for the three examples following this discussion. The first block of data is a reprint of the input where the following correlation between symbols may be observed:

Symbol	Corresponding value
ET	e_T
EI	e_I
AI	a_I
OMEGA I	Ω_I
OMI	ω_I
II	i_I
PHIO	ϕ_O
NUO1	ν_O (first)
NUOL	ν_O (last)
MU	μ
AT	a_T (ft)
TPER	P_T (sec)
VCT	V_{CT} (ft/sec)
DTTI	Δt (first)
DTTF	Δt (last)
DTWI	Δt_W (first)
DTWF	ΔT_W (last)
RMIN	r_{min}
H	h

In the case that the input is in the form of relative coordinate data, these Keplerian quantities for the initial interceptor orbit are computed from the relative coordinate data as has been explained. An example of this printout is shown in table II where a fourth row of variables is noted and XO, YO, and ZO correspond to x_O'', y_O'', and z_O'' and XODT, YODT, and ZODT correspond to $\dot{x}_O''$, $\dot{y}_O''$, and $\dot{z}_O''$.

After this output which records the data for the terminal orbits, the data are computed for the rendezvous problem and the following description of the printout nomenclature should be noted:

Symbol	Corresponding value
NUO	ν_o
DELTTT	Δt
DELTTW	Δt_w
PHIF	ϕ_f
NUI	ν_i
E	e
A	a
I	i
THI	θ_i
DELTTH	$\Delta\theta$
RHOM	ρ_{min}
DV1X	Component of $\Delta\hat{V}_i$ along $\hat{I}$
DV1Y	Component of $\Delta\hat{V}_i$ along $\hat{J}$
DV1Z	Component of $\Delta\hat{V}_i$ along $\hat{K}$
DV2X	Component of $\Delta\hat{V}_f$ along $\hat{I}$
DV2Y	Component of $\Delta\hat{V}_f$ along $\hat{J}$
DV2Z	Component of $\Delta\hat{V}_f$ along $\hat{K}$
DELV1	Magnitude of $\Delta\hat{V}_i$
DELV2	Magnitude of $\Delta\hat{V}_f$
DELV	Magnitude of $\Delta\hat{V}$

EXAMPLES

Example 1 – Interplanetary Transfer Between Massless Planets

To compute velocity increments required for interplanetary trajectories the input for the planet ephemeris is needed. This input can be found from reference 6. As an example, consider a trip from Earth to Mars which is to be accomplished by a two-impulse orbit. As the observation time t_o, January 4, 1964 (Julian date, 243-8398.5) is selected. From the 1964 ephemeris tables (pp. 18, 50, 172, 176), data for this problem are found which are referenced to the ecliptic plane. From the geometry of the ecliptic coordinate system (see refs. 1 and 2) and the definitions presented here, the following program input is determined:

$$e_T = 0.093372 \qquad \nu_o = 0.37^o$$

$$a_T = 7.4737 \times 10^{11} \qquad \Omega_I = 253.88^o$$

$\phi_o = 324.4^o$ $\omega_I = 233.02^o$

$e_I = 0.0167242$ $i_I = 1.850^o$

$a_I = 0.656301$

In order to be notified in the output whether the transfer arc is less than 0.6 astronomical units, let $r_{min} = 0.6a_I$. Also let

$$\mu = 4.679 \times 10^{21}\ ft^3/sec^2$$

$$h = 3.0$$

then

$$r_{min} = 0.393781$$

Since ν_o is not to be incremented (because it is fixed by the physics of the solar system), let ν_o (first), ν_o (last) be ν_o as shown and the increment $\delta\nu_o = 0$. Let it be desirable to collect data in increments of 20 days with transfer times from 160 to 260 days and waiting times of 0 to 40 days, and then use

$$\Delta t(\text{first}) = 160.0 \quad \Delta t(\text{last}) = 260.0 \quad \delta t = 20.0$$

$$\Delta t_w(\text{first}) = 0.0 \quad \Delta t_w(\text{last}) = 40.0 \quad \delta t_w = 20.0$$

If the data are to be analyzed with Δt as the primary independent variable and Δt_w as the parameter, Δt is incremented first. Therefore, set GUIDE at 2.0. The times are to be input and output in days so that OPTION is 3.0, and since Keplerian input is used, set DATAS as 1.0. Some output data are given in table I with time in days and all other quantities except μ, a_T, P_T, and V_{CT} are nondimensional.

Example 2 – Earth-Orbit Rendezvous

There are many approximate schemes which have been derived and investigated for determining the velocity increments required to rendezvous between similar orbits. Such investigations have been mainly concerned with circular orbits or first-order representations of elliptic orbits (near circular orbits). The simplest case of rendezvous with a circular target orbit (as done by Clohessy and Wiltshire, ref. 7) gives a closed-form solution to the linearized equations of motion for the velocity increment required to rendezvous in a given amount of time. This linearized solution is accurate for the terminal phases of a general rendezvous problem but the accuracy is soon lost for large initial interceptor-target ranges. It should be noticed that even in the docking phase, this solution may not be too accurate if the target orbit is highly eccentric. Further attempts to increase the applicability of the Clohessy-Wiltshire results usually require an iteration scheme (Anthony and Sasaki, ref. 8) or some other process to get a solution of the velocity increment required to rendezvous.

It is noted that the solution of the exact equations presented herein is not too involved once a computer program is available, and, of course, offers the advantage of giving correct results for both large eccentricities and inclinations.

As an example of the use of this program for earth-orbit rendezvous, assume that the following elements are used for the target orbit:

$$e_T = 0.0234$$

$$a_T = 2.248 \times 10^7 \text{ ft}$$

Also let

$$\phi_o = 0.0^o$$

$$r_{min} = 0.954$$

$$\mu = 1.408 \times 10^{16} \text{ ft}^3/\text{sec}^2$$

$$h = 3.0$$

The interceptor position and velocity are given in relative coordinates:

$$x''_o = -0.01692 \qquad \dot{x}''_o = -0.00376$$

$$y''_o = 0.0376 \qquad \dot{y}''_o = 0.1526$$

$$z''_o = 0.0 \qquad \dot{z}''_o = 0.0$$

Note that e_I, a_I, ν_o, i_I, Ω_I, and ω_I are computed from these relations. Let the time be input and output in minutes:

$$\Delta t(\text{first}) = 2.0 \qquad \Delta t(\text{last}) = 40.0 \qquad \delta t = 10.0$$

$$\Delta t_w(\text{first}) = 0.0 \qquad \Delta t_w(\text{last}) = 30.0 \qquad \delta t_w = 10.0$$

If a plot of Δt_w as abscissa and Δt as a constant parameter is desired, set GUIDE at 1.0. Since $\frac{\text{Input}}{\text{Output}}$ time is in minutes, OPTION is 2.0, and since relative coordinate data are used, set DATAS at 2.0.

The output of the computer program for this input is given in table II.

Example 3 – Transfer Orbits

The problem of orbital transfer between two arbitrary elliptic orbits may also be accomplished by this computational procedure. The use of this method in orbital transfer problems is indicated in this example where the minimum two-impulse transfer orbit is desired between two fixed terminal orbits. The transfer orbit requiring minimum velocity increment may be found by proper interpretation of the data and the use of a plotter. The program as presented here does not have a gradient or other optimization scheme to find the minimum velocity transfer directly. However, an example is shown of one way of obtaining the optimum (two-impulse) orbital transfer.

For the terminal orbits the following input data are used:

$$e_T = 0.5 \qquad a_T = 1.0 \quad \text{(arbitrary)}$$

$$\Omega_I = 90^\circ \qquad e_I = 0.2$$

$$i_I = 30^\circ \qquad a_I = 0.9$$

$$\omega_I = -90^\circ$$

Also let

$$r_{min} = 0.45$$

$$h = 3.0$$

$$\mu = 1.0 \quad \text{(arbitrary)}$$

The true anomaly ν_o of the interceptor at $t = t_o$ is chosen in the following range:

$$\nu_o(\text{first}) = 0.0$$

$$\nu_o(\text{last}) = 350^\circ$$

$$\delta\nu_o = 10^\circ$$

which is incremented automatically by the program. To obtain all the possible transfer arcs, it is necessary to increment either Δt_W or ν_o, but not both. As ν_o is chosen to be the incremented parameter, the wait time is set equal to zero.

The true anomaly ϕ_o is chosen, and when the transfer time Δt is varied, a set of curves for the variation of ΔV with Δt with ν_o as the parameter are obtained for a specified value of ϕ_o. The program does not increment ϕ_o automatically as it does for Δt, Δt_W, and ν_o. However, it always returns to read the first data card of a new set after running through one complete set of data. If there are no more sets of data, the computation ends; if there is another complete data set, it starts anew. To increment ϕ_o, it is necessary to reproduce all data cards for every new value of ϕ_o and to place these one behind the other. (See appendix C.)

As set out above, each complete data set gives one complete set of curves for a given ϕ_o with ν_o as the parameter. (See sketch 1.)

As an aid, sketch 1 has been drawn with ϕ_f as the abscissa value rather than Δt and is possible because of the functional relation between Δt and ϕ_f or

$$\phi_f = \phi_o + F(\Delta t)$$

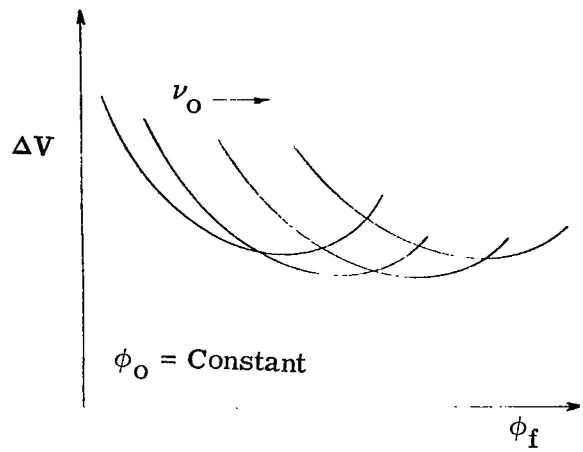
since $\Delta t_w = 0$.

The transfer time Δt is read in as follows:

$$\Delta t(\text{first}) = 0.09$$

$$\Delta t(\text{last}) = 1.008$$

$$\delta t = 0.009$$



Sketch 1

These quantities are dimensionless ratios to the target orbit period. The waiting time is input as

$$\Delta t_w(\text{first}) = \Delta t_w(\text{last}) = \delta t_w = 0$$

Some sample computer output is shown in table III for $\phi_o = 0.0^\circ$.

From a Beckman automatic plotter, the characteristics of one of the lower ΔV transfer orbits are found to be

$\nu_i = 55^\circ$	$\Delta\theta = 132.47^\circ$
$\phi_f = 193.58^\circ$	$i = -22.88^\circ$
$\Delta t = 0.64$	$\rho_{min} = 0.78$
$e = 0.4361$	$\Delta V_i = 0.0808$
$a = 1.086$	$\Delta V_f = 0.2563$
$\theta_i = 71.96^\circ$	$\Delta V = 0.3371$

CONCLUDING REMARKS

This report presents a technique for solving three-dimensional orbital problems in a straightforward manner using the exact solutions or Kepler solutions to the equations of motion. Basically the method developed is an iteration on the Kepler equation using the true anomaly as the iteration parameter and the mean anomaly or transfer time compared with a prespecified transfer time as the stopping criteria. To aid in the choice of the true anomaly to begin the iteration, certain boundaries are devised within which the solution must lie. The iteration is performed directly on the Kepler equations and no derivatives are necessary. This method works very well and the computation time compares favorably with other methods, the typical run times being about 0.003 minute per transfer. The generality of the program format presented allows rapid computations, with simple engineering input parameters, of interplanetary rendezvous or near-planet rendezvous cases. For instance the program has proven useful in studies concerning fuel requirements for abort missions during lunar letdown of the lunar excursion module and command module.

Langley Research Center,
National Aeronautics and Space Administration,
Langley Station, Hampton, Va., February 14, 1966.

APPENDIX A

OBTAINING KEPLERIAN ORBITAL ELEMENTS FROM RELATIVE COORDINATE DATA

In many problems concerning rendezvous, the initial conditions are given in terms of the relative position and velocity of the interceptor with respect to the target. If the Keplerian equations are used for the exact solution of the motion, the Keplerian elements in terms of the relative coordinates must be obtained. Let x''_o, y''_o, z''_o be the relative position vector (see fig. 5) of the interceptor with respect to the x'', y'', z'' coordinate system attached to the target; x'' points away from the gravitational center, y'' is in the positive direction of angular motion, and z'' is along Z normal to the $x''y''$ plane.

Suppose that the following are given in dimensionless form: e_T, ϕ_o, x''_o, y''_o, z''_o, $\dot{x}''_o$, $\dot{y}''_o$, and $\dot{z}''_o$. It is desired to find the Keplerian constants e_I, a_I, ν_o, Ω_I, ω_I, and i_I. By using the formal application of vector analysis, let $\hat{i},\hat{j},\hat{k}$ be unit vectors along x'', y'', and z'', and obtain the radius vector

$$\hat{\xi} = (r + x'')\hat{i} + y''\hat{j} + z''\hat{k} \tag{A1}$$

for the interceptor, and since the rotational rate of the (x'',y'',z'') coordinate system is $\dot{\phi}\hat{k}$, the velocity is

$$\dot{\hat{\xi}} = (\dot{r} + \dot{x}')\hat{i} + \dot{y}'\hat{j} + z'\hat{k} + \dot{\phi}\hat{k} \times \hat{\xi} \tag{A2}$$

where

$$\hat{k} \times \hat{\xi} = (r + x'')\hat{j} - y''\hat{i} \tag{A3}$$

The velocity of the interceptor becomes:

$$\hat{V} \equiv \dot{\hat{\xi}} = (\dot{r} + \dot{x}'' - \dot{\phi}y'')\hat{i} + (\dot{y}'' + \dot{\phi}(r + x''))\hat{j} + \dot{z}''\hat{k} \tag{A4}$$

The fundamental relations of orbital mechanics show that

$$\left.\begin{aligned}
p_T &= 1 - e_T{}^2 \\
r &= \frac{p_T}{1 + e_T \cos \phi} \\
\dot{r} &= \frac{e_T \sin \phi}{\sqrt{p_T}} \\
\dot{\phi} &= \frac{\sqrt{p_T}}{r^2}
\end{aligned}\right\} \tag{A5}$$

From these expressions the velocity of the interceptor in its initial orbit can be written in terms of the desired quantities along the axes of the moving x'', y'', z'' coordinate system. Quantities considered here are at the time $t = 0$ and are subscripted accordingly. The velocity components at $t = t_o$ along the x'', y'', and z'' axes will be denoted by V_{o1}, V_{o2}, and V_{o3}, respectively; thus, the interceptor velocity becomes

$$\hat{V} = V_{o1}\hat{i} + V_{o2}\hat{j} + V_{o3}\hat{k} \tag{A6}$$

By using equations (A4), (A5), and (A6), the components are written in terms of known quantities as

$$\left.\begin{aligned} V_{o1} &= \frac{e_T \sin \phi_o}{\sqrt{p_T}} + \dot{x}''_o - y''_o \frac{\sqrt{p_T}}{r_o^2} \\ V_{o2} &= \dot{y}''_o + \left(r_o + x''_o\right)\frac{\sqrt{p_T}}{r_o^2} \\ V_{o3} &= \dot{z}''_o \end{aligned}\right\} \tag{A7}$$

The energy equation for the interceptor at time t_o may be written (nondimensionally) as

$$V_o^2 - \frac{2}{\xi_o} = -\frac{1}{a_I} \tag{A8}$$

where ξ_o, V_o are magnitudes of the defined vectors $\hat{\xi}$, $\hat{V}$ at time t_o or

$$\xi_o = \sqrt{\left(r_o + x''_o\right)^2 + \left(y''_o\right)^2 + \left(z''_o\right)^2} \tag{A9}$$

$$V_o = \sqrt{V_{o1}^2 + V_{o2}^2 + V_{o3}^2} \tag{A10}$$

By using equation (A8), solve for the semi-major axis a_I

$$a_I = \frac{\xi_o}{2 - \xi_o V_o^2} \tag{A11}$$

To determine the eccentricity of the initial interceptor orbit, write the angular momentum at time t_o of the interceptor,

$$\hat{H}_o = \hat{\xi}_o \times \hat{V}_o \tag{A12}$$

where $\hat{\xi}_o$ and $\hat{V}_o$ are dimensionless quantities. If

$$\hat{H}_o = H_{o1}\hat{i} + H_{o2}\hat{j} + H_{o3}\hat{k}$$

performing the indicated operations yields

$$\left.\begin{aligned} H_{o1} &= y_o''V_{o3} - z_o''V_{o2} \\ H_{o2} &= z_o''V_{o1} - \left(r_o + x_o''\right)V_{o3} \\ H_{o3} &= \left(r_o + x_o''\right)V_{o2} - y_o''V_{o1} \end{aligned}\right\} \tag{A13}$$

and

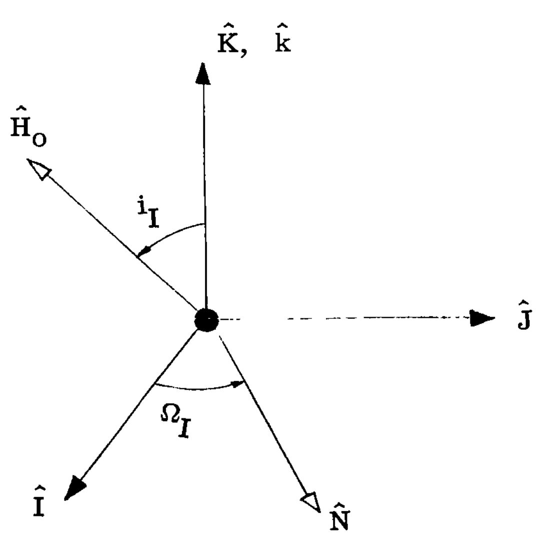
$$H_o = \sqrt{\left(H_{o1}\right)^2 + \left(H_{o2}\right)^2 + \left(H_{o3}\right)^2} \tag{A14}$$

In the nondimensional form, the angular momentum and semi-latus rectum may easily be shown to be related as

$$p_I = H_o^{\,2} \tag{A15}$$

and therefore the well-known relationship $p_I = a_I\left(1 - e_I^{\,2}\right)$ gives with equation (A15):

$$e_I = \sqrt{1 - \frac{H_o^{\,2}}{a_I}} \tag{A16}$$



Sketch 2

Sketch 2 and figure 5 are given as aids in describing how the angular measures i_I, Ω_I, and ω_I are determined. The unit vector $\hat{N}$ directed along the line of nodes of the initial interceptor orbit and pointing toward the ascending node is defined by the relation

$$\hat{k} \times \hat{H}_o = \left|H_o \sin i_I\right|\hat{N} \tag{A17}$$

The inclination i_I is also described by the relation

$$\hat{k} \cdot \hat{H}_o = H_o \cos i_I \tag{A18}$$

and equation (A18) gives

$$i_I = \cos^{-1} \frac{H_{o3}}{H_o} \qquad (0 < i_I < \pi) \tag{A19}$$

From figure 5, it is geometrically evident that

$$\hat{I} = \hat{i} \cos \phi_o - \hat{j} \sin \phi_o \tag{A20}$$

and also from the definition of the angle Ω_I

$$\cos \Omega_I = \hat{I} \cdot \hat{N} \tag{A21}$$

$$\hat{k} \sin \Omega_I = \hat{I} \times \hat{N} \tag{A22}$$

and expanding the vector manipulation of equation (A17) gives

$$\hat{N} = \frac{H_{o1}}{H_o \sin i_I} \hat{j} - \frac{H_{o2}}{H_o \sin i_I} \hat{i} \tag{A23}$$

From the well-known relationship

$$\xi_o = \frac{p_I}{1 + e_I \cos \nu_o}$$

obtain

$$\cos \nu_o = \frac{1}{e_I} \frac{p_I}{\xi_o} - 1 \tag{A24}$$

A basic orbital relationship which is of value here is

$$\sin \nu_o = \frac{\sqrt{p_I}}{e_I} \dot{\xi}_o \tag{A25}$$

and, as $\dot{\xi}_o$ is the component of $\hat{V}_o$ along $\hat{\xi}_o$,

$$\dot{\xi}_o = \hat{V}_o \cdot \frac{\hat{\xi}_o}{\xi_o} \tag{A26}$$

In order to find the angle ω_I, note that another geometric relation is

$$\hat{\xi}_o \cdot \hat{N} = \xi_o \cos (\omega_I + \nu_o) \tag{A27}$$

and also

$$\hat{N} \times \hat{\xi}_o = \xi_o \sin\left(\omega_I + \nu_o\right)\frac{\hat{H}_o}{H_o} \tag{A28}$$

Note also that ν_o may be found from equations (A24) and (A25) as

$$\nu_o = \tan^{-1}\left(\frac{\sin \nu_o}{\cos \nu_o}\right) \qquad \left(0 \leqq \nu_o < 2\pi\right) \tag{A29}$$

From equations (A20), (A21), and (A23),

$$\cos \Omega_I = -\frac{H_{o2} \cos \phi_o + H_{o1} \sin \phi_o}{H_o \sin i_I} \tag{A30}$$

and the proper vector manipulations with equations (A20), (A22), and (A23) yield

$$\sin \Omega_I = \frac{H_{o1} \cos \phi_o - H_{o2} \sin \phi_o}{H_o \sin i_I} \tag{A31}$$

From equations (A30) and (A31),

$$\Omega_I = \tan^{-1}\left(\frac{\sin \Omega_I}{\cos \Omega_I}\right) \qquad \left(0 \leqq \Omega_I < 2\pi\right) \tag{A32}$$

Similarly, performing the indicated vector manipulations on equations (A27) and (A28) leads to the scalar equations

$$\cos\left(\omega_I + \nu_o\right) = \frac{\xi_{o2}H_{o1} - \xi_{o1}H_{o2}}{\xi_o H_o \sin i_I} \tag{A33}$$

$$\sin\left(\omega_I + \nu_o\right) = \frac{\xi_{o3}}{\xi_o \sin i_I} \tag{A34}$$

If the inclination i_I is zero, Ω_I is arbitrary; thus, let

$$\Omega_I = 0 \tag{A35}$$

Also, equations (A27) and (A28) yield

$$\cos\left(\omega_I + \nu_o\right) = \frac{\xi_{o1} \cos \phi_o - \xi_{o2} \sin \phi_o}{\xi_o} \tag{A36}$$

$$\sin\left(\omega_I + \nu_o\right) = \frac{\xi_{o1} \sin \phi_o + \xi_{o2} \cos \phi_o}{\xi_o} \tag{A37}$$

Whether i_I is zero or not, equations (A33), (A34), or (A36), (A37) give

$$\omega_I + \nu_o = \tan^{-1}\left[\frac{\sin\left(\omega_I + \nu_o\right)}{\cos\left(\omega_I + \nu_o\right)}\right] \tag{A38}$$

and equation (A38) with equation (A29) give the desired expression

$$\omega_I = \left(\omega_I + \nu_o\right) - \nu_o \tag{A39}$$

Equations (A11), (A16), (A19), (A29), (A32), and (A39) give the information necessary to compute the desired Keplerian elements.

APPENDIX B

MATHEMATICAL DESCRIPTION OF PROBLEM

The following description is a logical flow of the problem as it is programed.

Description of Initial and Final Properties

If the initial and terminal states of an orbit referred to two other elliptic orbits, namely, the interceptor initial orbit and the target orbit, and also the time required to transfer from orbit to orbit are known, the velocity increments required to establish the transfer orbit and the elements of this transfer orbit may be computed. The initial and terminal states may be found from the input as follows. Suppose that the following data are given: μ, a_T, e_T, ϕ_o, e_I, a_I, ω_I, Ω_I, i_I, ν_o, Δt_W, and Δt. Recall that all quantities except μ and a_T are dimensionless.

Compute the semi-latus rectum of the initial interceptor orbit and the target orbit from

$$p_I = a_I\left(1 - e_I^2\right) \tag{B1}$$

$$p_T = 1 - e_T^2 \tag{B2}$$

The target orbital period in seconds is

$$P_T = 2\pi\sqrt{\frac{a_T^3}{\mu}} \tag{B3}$$

The eccentric anomaly of the target at $t = t_o$ is found from e_T and ϕ_o by the defined relationship

$$\tan\frac{E_{To}}{2} = \sqrt{\frac{1 - e}{1 + e}}\tan\frac{\phi_o}{2}$$

and then functionally as

$$E_{To} = F\left(e_T, \phi_o\right) \tag{B4}$$

For computational purposes, define the function $F(\alpha,\beta)$ as

$$F(\alpha,\beta) = 2n\pi + 2\tan^{-1}\left(\sqrt{\frac{1 - \alpha}{1 + \alpha}}\gamma\tan\frac{\beta}{2}\right)_{1^{st}\text{ and }4^{th}\text{ quadrants}} \tag{B5a}$$

where

$$\left.\begin{aligned} \alpha &= e_T \\ \beta &= \phi_o \end{aligned}\right\} \tag{B5b}$$

and

$$n = \text{Integer}\left(\frac{\beta + \pi}{2\pi}\right)_{\text{lowest integer value}} \tag{B5c}$$

$$\left.\begin{aligned} \gamma &= 1.0 \qquad (\alpha < 1.0) \\ \gamma &= -1.0 \qquad (\alpha > 1.0) \end{aligned}\right\} \tag{B5d}$$

Note that allowance is made here for the possibility of hyperbolic orbits because $F(\alpha,\beta)$ will also be used for transfer orbits.

The mean anomaly of the target orbit at $t = t_o$ is found directly from Kepler's equation

$$M_{To} = E_{To} - e_T \sin E_{To} \tag{B6}$$

The mean anomaly of the target at the final time $t = t_f$ is

$$M_{Tf} = M_{To} + 2\pi\left(\Delta t + \Delta t_w\right) \tag{B7}$$

The eccentric anomaly of the target at $t = t_f$ is required so that the true anomaly at $t = t_f$ may ultimately be obtained. It is possible to solve for the eccentric anomaly by use of an expansion in terms of the infinite Bessels series (see ref. 9) but it was found to be more rapid to iterate the Kepler equation by using the method of differential correction as shown in reference 5 or 9. The steps in the iteration procedure are as follows:

(1) Estimate the initial value E_1 from truncated series solution of the Kepler equation

$$E_1 = M_o + e \sin M_o + \frac{1}{2} e^2 \sin 2M_o \qquad \left(M_o = M_{Tf};\quad e = e_T\right) \tag{B8}$$

(2) Compute the following sequence:

$$i = 1$$
$$M_i = E_i - e_T \sin E_i$$
$$\Delta M_i = M_{Tf} - M_i$$
$$\Delta E_i = \Delta M_i \Big/ \left(1 - e_T \cos E_i\right) \qquad i = i + 1$$
$$E_{i+1} = E_i + \Delta E_i$$
$$\text{Compare } E_{i+1} : E_i \longrightarrow \begin{cases} E_{i+1} \neq E_i \\ E_{i+1} = E_i \end{cases}$$

which gives after the completed iteration:

$$E_{Tf} = E_{i+1} \tag{B9}$$

At $t = t_i$ the eccentric anomaly of the interceptor E_{Ii} is obtained from ν_o, e_I, and Δt_w in the same manner as E_{Tf}. The expressions are rewritten here for completeness with eccentric anomaly of interceptor at $t = t_o$ as

$$E_{Io} = F\left(e_I, \nu_o\right)$$

where $F\left(e_I, \nu_o\right)$ is the function previously defined (eq. (B5)) with $\alpha = e_I$ and $\beta = \nu_o$. Also

$$M_{Io} = E_{Io} - e_I \sin \left(E_{Io}\right)$$

At $t = t_i$ the interceptor mean anomaly is found from

$$M_{Ii} = M_{Io} + 2\pi \sqrt{\frac{1}{a_I^3}}\, \Delta t_w$$

The iteration steps are shown in equations (B8) and (B9) with the following replacements: e_T by e_I; M_{Tf} by M_{Ii}; E_{Tf} by E_{Ii}.

With this information the true anomalies at the initial and final times of the interceptor in its initial orbit and the target in its orbit may be computed. The routine given by equations (B4) and (B5) will give these values if $-\alpha$ is replaced by the corresponding e. Thus, the true anomaly of the interceptor in its initial orbit at $t = t_i$ is, α and β in F being replaced by the parameters $-e_I$ and E_{Ii}

$$\nu_i = F\left(-e_I, E_{Ii}\right) \tag{B10}$$

and the target in its plane at $t = t_f$ is

$$\phi_f = F\left(-e_T, E_{Tf}\right) \tag{B11}$$

With these quantities it is possible to compute the properties ξ_i, $\dot{\xi}_i$, $\dot{\nu}_i$, r_f, $\dot{r}_f$, and $\dot{\phi}_f$ which will be required for computation of the velocity increments. The initial radius vector is

$$\xi_i = \frac{p_I}{1 + e_I \cos \nu_i} \tag{B12}$$

The rate of change of true anomaly at $t = t_i$ is

$$\dot{\nu}_i = \sqrt{p_I}\,\frac{1}{\left(\xi_i\right)^2} \tag{B13}$$

Also the rate of change of the radial distance is

$$\dot{\xi}_i = \frac{e_I \sin \nu_i}{\sqrt{p_I}} \tag{B14}$$

For the final time $t = t_f$, the properties of the target orbit are

$$r_f = \frac{p_T}{1 + e_T \cos \phi_f} \tag{B15}$$

$$\dot{\phi}_f = \frac{\sqrt{p_T}}{r_f^2} \tag{B16}$$

$$\dot{r}_f = \frac{e_T \sin \phi_f}{\sqrt{p_T}} \tag{B17}$$

If an interceptor is to transfer (or rendezvous) from one orbit to another, the two orbits being determined by this information, it must begin at $t = t_i$ in the initial orbit, change velocity (instantaneously) to the transfer orbit at $t = t_i$, and travel until $t = t_f$ in the transfer orbit. Then it instantaneously changes its velocity to satisfy the terminal or target orbit properties. Enough information is now available to compute the initial and final terminal velocities but further discussion of this procedure will be delayed until the initial and final transfer orbit properties have been computed.

Computation of Transfer Orbit Properties

Information giving initial and final terminal vectors in space $\hat{\xi}_i$, $\hat{r}_f$ is available, and it is now necessary to determine the transfer conic which connects these two vectors in the specified transfer time Δt. Reference 3 shows that the solution to this problem is unique. However, because of the implicit nature of the Kepler problem, it is necessary to perform an iteration in order to obtain the solution. The method used here is a straightforward iteration of the Keplerian orbital equations as was also described in reference 3.

The plane of the transfer orbit is specified by the plane containing the initial and final position vectors and the center of attraction. The transfer plane inclination i and the transfer angle $\Delta\theta$ are determined with the aid of vector representation. Let $\hat{I}$, $\hat{J}$, and $\hat{K}$ be unit vectors along X, Y, and Z axes; let $\hat{i}$, $\hat{j}$, and $\hat{k}$ be unit vectors along x, y, and z axes as shown in figures 1, 2, and 3. Let $\left[a_{ij}\right]$ be the x to X transformation matrix so that a general vector $\hat{A}$ transforms as:

$$\hat{A}(IJK) = \left[a_{ij}\right]\hat{A}(ijk)$$

The initial vector may be written as

$$\hat{\xi}_i = \xi_i\left(\cos \nu_i\hat{i} + \sin \nu_i\hat{j}\right) \tag{B18}$$

By letting $\xi_{i1},\xi_{i2},\xi_{i3}$ be the components of $\hat{\xi}_i$ along the X,Y,Z axes, the following relations are obtained:

$$\hat{\xi}_i = \xi_{i1}\hat{I} + \xi_{i2}\hat{J} + \xi_{i3}\hat{K}$$

$$\hat{r}_f = r_f\left(\cos \phi_f\hat{I} + \sin \phi_f\hat{J}\right)$$

In order to examine the vectors in the $\hat{I},\hat{J},\hat{K}$ system, a transformation of components is required; thus,

$$\hat{\xi}_i(I,J,K) = \left[a_{ij}\right]\hat{\xi}_i(ijk) \tag{B19}$$

where

$$\left[a_{ij}\right] = \begin{bmatrix} \cos \omega_i \cos \Omega_I - \cos i_I \sin \Omega_I \sin \omega_I & -\sin \omega_I \cos \Omega_I - \cos i_I \sin \Omega_I \cos \omega_I & \sin \Omega_I \sin i_I \\ \cos \omega_I \sin \Omega_I + \cos i_I \cos \Omega_I \sin \omega_I & -\sin \omega_I \sin \Omega_I + \cos i_I \cos \Omega_I \cos \omega_I & -\cos \Omega_I \sin i_I \\ \sin i_I \sin \omega_I & \sin i_I \cos \omega_I & \cos i_I \end{bmatrix} \tag{B20}$$

From equations (B18), (B19), and (B20),

$$\left.\begin{aligned}
\xi_{i1} &= \xi_i\left[\cos \Omega_i \cos\left(\nu_i + \omega_I\right) - \cos i_I \sin \Omega_I \sin\left(\nu_i + \omega_I\right)\right] \\
\xi_{i2} &= \xi_i\left[\sin \Omega_I \cos\left(\nu_i + \omega_I\right) + \cos i_I \cos \Omega_I \sin\left(\nu_i + \omega_I\right)\right] \\
\xi_{i3} &= \xi_i\left[\sin i_I \sin\left(\nu_i + \omega_I\right)\right]
\end{aligned}\right\} \tag{B21}$$

To determine $\Delta\theta$ and i, note the vector identities:

$$\hat{\xi}_i \cdot \hat{r}_f = \xi_i r_f \cos \Delta\theta \tag{B22}$$

$$\hat{\xi}_i \times \hat{r}_f = \xi_i r_f \sin \Delta\theta \hat{k}' \tag{B23}$$

Note that the unit vector $\hat{k}'$ along the z'-axis is written as

$$\hat{k}' = \hat{I} \sin i \sin \Omega - \hat{J} \sin i \cos \Omega + \hat{K} \cos i \tag{B24}$$

and since

$$\Omega + \pi = \phi_f$$

$$\hat{k}' = -\sin i \sin \phi_f \hat{I} + \sin i \cos \phi_f \hat{J} + \cos i \hat{K}$$

From equations (B18), (B19), (B23), and (B24) find

$$\cos \Delta\theta = \frac{\xi_{i1} \cos \phi_f + \xi_{i2} \sin \phi_f}{\xi_i}$$

$$\sin \Delta\theta \sin i = \frac{\xi_{i3}}{\xi_i}$$

$$\tan i = \frac{\xi_{i3}}{\xi_{i1} \sin \phi_f - \xi_{i2} \cos \phi_f}$$

Combining these equations and solving for i yields the following expression:

$$i = \arc\tan \frac{\xi_{i3}}{\xi_{i1} \sin \phi_f - \xi_{i2} \cos \phi_f} \qquad \left(-\frac{\pi}{2} < i < \frac{\pi}{2}\right) \tag{B25}$$

and also

$$\cos \Delta\theta = \frac{\xi_{i1} \cos \phi_f + \xi_{i2} \sin \phi_f}{\xi_i} \tag{B26}$$

$$\sin \Delta\theta = \frac{\xi_{i1} \sin \phi_f - \xi_{i2} \cos \phi_f}{\xi_i \cos i} \tag{B27}$$

$$\Delta\theta = \arc\tan \frac{\sin \Delta\theta}{\cos \Delta\theta} \qquad (0 \leqq \Delta\theta \leqq 2\pi) \tag{B28}$$

The nodal angle Ω may be expressed by

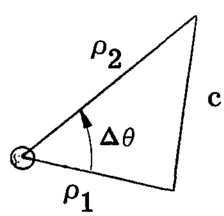
$$\left.\begin{aligned} \Omega &= \phi_f + \pi & (i > 0) \\ \Omega &= \phi_f & (i < 0) \end{aligned}\right\} \tag{B29}$$

An inherent symmetry in the equations exists so that it makes no difference in the iteration whether the transfer from $\hat{\xi}_i$ to $\hat{r}_f$ or $\hat{r}_f$ to $\hat{\xi}_i$ is considered, and advantage of this symmetry is taken by considering transfers only from the shortest of $\hat{\xi}_i$, $\hat{r}_f$ to the longest. Therefore, let ρ_1 be the minimum of ξ_i, r_f and ρ_2 be the maximum of ξ_i, r_f. (If $r_f = \xi_i$, $\rho_1 = \rho_2 = \rho = r_f = \xi_i$.)

The chord c is given by

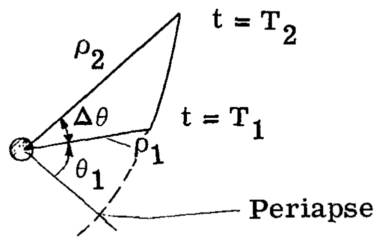
$$c = \sqrt{\rho_1^2 + \rho_2^2 - 2\rho_1\rho_2 \cos \Delta\theta} \tag{B30}$$

A geometric picture is shown in the following sketch:



Sketch 3.- The quantities ρ_1, ρ_2, $\Delta\theta$, and c.

By assuming a periapse angle of θ_1 in the transfer plane, an orbit between ρ_1 and ρ_2 can be determined as shown in the following sketch:



Sketch 4.- Transfer orbit for corresponding θ_1.

For each properly chosen angle θ_1, there is a conic section which passes through ρ_1,ρ_2 and an associated time of passage T_1,T_2. Hence, for a given θ_1, a given transfer time $T_2 - T_1$ is generated. In general, this time will not be the desired transfer time Δt, and thus an iteration is necessary. The necessary calculations for the determination of the orbit by this method are now developed. Define

$$\theta_2 = \theta_1 + \Delta\theta \tag{B31}$$

By eliminating p from the well-known expressions

$$\left.\begin{aligned} \rho_1 &= \frac{p}{1 + e \cos \theta_1} \\ \rho_2 &= \frac{p}{1 + e \cos \theta_2} \end{aligned}\right\} \tag{B32}$$

the unknown eccentricity of the transfer orbit is found from

$$e = \frac{\rho_2 - \rho_1}{\rho_1 \cos \theta_1 - \rho_2 \cos \theta_2} \tag{B33}$$

and the semilatus rectum

$$p = \rho_1\left(1 + e \cos \theta_1\right) \tag{B34}$$

The semi-major axis is computed as

$$a = \frac{p}{1 - e^2} \tag{B35}$$

All the orbital parameters may be fixed by choice of the periapse angle θ_1.

A distinctive value of θ_1 is the periapse angle θ_p for parabolic transfer from ρ_1 to ρ_2 and the associated parabolic transfer time $\Delta T_p = T_2 - T_1$. This value of θ_1 is easily found since $e = 1.0$ for parabolic orbits. Letting $\theta_1 = \theta_p$ and using equation (B32) yields

$$\rho_1\left(1 + \cos\ \theta_p\right) = \rho_2\left(1 + \cos\ \theta_2\right) \equiv \rho_2\left[1 + \cos\left(\Delta\theta + \theta_p\right)\right] \tag{B36}$$

or

$$\left(\rho_1 - \rho_2 \cos\ \Delta\theta\right)\cos\ \theta_p + \left(\rho_2 \sin\ \Delta\theta\right)\sin\ \theta_p = \rho_2 - \rho_1$$

Then θ_p is found by trigonometric identity

$$\sin\left(\theta_p + \psi\right) = \frac{\rho_2 - \rho_1}{c} > 0 \tag{B37a}$$

where the angle ψ is defined by

$$\sin\ \psi = \frac{\rho_1 - \rho_2 \cos\ \Delta\theta}{c}$$

$$\cos\ \psi = \frac{\rho_2 \sin\ \Delta\theta}{c}$$

$$\psi = \tan^{-1}\left(\frac{\sin\ \psi}{\cos\ \psi}\right) \qquad \left(-\frac{\pi}{2} \leqq \psi \leqq \frac{3\pi}{2}\right)$$

Let

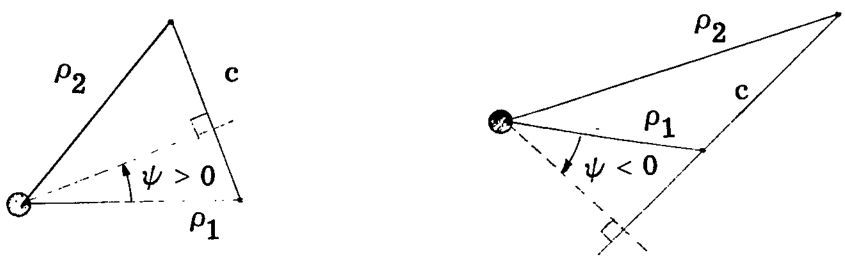
$$\bar{\alpha} = \sin^{-1}\left(\frac{\rho_2 - \rho_1}{c}\right) \qquad \left(0 \leqq \bar{\alpha} \leqq \frac{\pi}{2}\right)$$

and then note both values of θ_p in an interval of 2π

$$\theta_p = \theta_p\Big|_{\text{minimum}} = \bar{\alpha} - \psi \tag{B37b}$$

$$\theta_p = \theta_p\Big|_{\text{maximum}} = \pi - (\bar{\alpha} + \psi) \tag{B37c}$$

A geometrical description of ψ is shown in the following sketch:



Sketch 5.- Geometric interpretation of ψ.

The semi-latus rectum corresponding to θ_p is

$$p_p = \rho_1\left(1 + \cos \theta_p\right)$$

The parabolic transfer time is then found as follows: The time to travel from periapse to ρ_1 is found in reference 10 and is

$$T_{p1} = \frac{1}{2\pi}\sqrt{p_p^{\,3}}\left(\frac{1}{6} \tan^3 \frac{\theta_p}{2} + \frac{1}{2} \tan \frac{\theta_p}{2}\right)$$

and from periapse to ρ_2 is

$$T_{p2} = \frac{1}{2\pi}\sqrt{p_p^{\,3}}\left(\frac{1}{6} \tan^3 \frac{\theta_p + \Delta\theta}{2} + \frac{1}{2} \tan \frac{\theta_p + \Delta\theta}{2}\right)$$

so the parabolic transfer time is

$$\Delta T_p = T_{p2} - T_{p1} \tag{B38}$$

This value may be used to compare the desired time Δt of transfer and to determine whether the orbit is elliptic $\left(\Delta t > \Delta T_p\right)$ or hyperbolic $\left(\Delta t < \Delta T_p\right)$.

There are certain definite regions in which θ_1 must remain for a solution to exist. These regions are found by use of equation (B33) and the positive sign of e and $\rho_2 - \rho_1$. These equations imply

$$\rho_1 \cos \theta_1 > \rho_2 \cos \theta_2$$

which leads to $\sin\left(\theta_1 + \psi\right) > 0$ or

$$\pi - \psi > \theta_1 > -\psi$$

where ψ is defined in equation (B37). It is seen that the lower limit for θ_1 is $-\psi$ for all cases. The angle θ_p is greater than $-\psi$ as it is given by equation (B37). For an arbitrary e, θ_p is replaced by θ_1 and equations (B37a) are written

$$\theta_1 = \sin^{-1}\left(\frac{\rho_2 - \rho_1}{ec}\right)\Bigg|_{\text{First quadrant}} - \psi \tag{B39}$$

From equations (B39) and (B37), the following inequalities are obtained:

$$\left.\begin{array}{ll} \theta_1 > \theta_p\big|_{\text{minimum}} & (e < 1) \\ \theta_1 < \theta_p\big|_{\text{minimum}} & (e > 1) \end{array}\right\} \tag{B40}$$

It may also be shown that the true maximum value for θ_1 is less than $\pi - \psi$ and is the second parabolic solution

$$\theta_1\big|_{\text{maximum}} = \theta_p\big|_{\text{maximum}} \tag{B41}$$

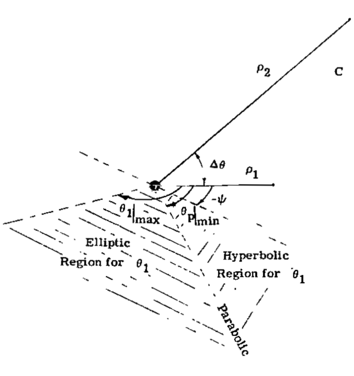
Hence, the following regions for choice of the iterative periapse angles θ_1 are determined from these relations:

(1) If $\Delta t < \Delta T_p$, then $e > 1$ (hyperbolic transfer) and $-\psi < \theta_1 < \theta_p$

(2) If $\Delta t > \Delta T_p$, then $e < 1$ (elliptic transfer) and $\theta_p < \theta_1 < \pi - 2(\psi + \theta_p)$

(3) If $\Delta t = \Delta T_p$, then $e = 1$ and the solution is $\theta_1 = \theta_p$

A geometric interpretation may be given to these regions and an example is shown in the following sketch:



Sketch 6.- Regions of possible choice for the periapse θ_1.

APPENDIX B

Elliptic transfer orbits.- In the case that $\Delta t > \Delta T_p$, the transfer orbit for rendezvous must be elliptic and hence $\theta_p < \theta_1 < \pi - 2\psi - \theta_p$. The iteration scheme used here is a straightforward computation of the values $T_2 - T_1$ for values of θ_1 starting with θ_p and continuing until $T_2 - T_1$ becomes equal to Δt. That is, the interval of θ_1 is divided into h smaller intervals $\delta\theta$ where

$$\delta\theta = \frac{\pi - 2(\psi + \theta_p)}{h} = \frac{\theta_p\big|_{max} - \theta_p\big|_{min}}{h}$$

The first estimation of θ_1 is $\theta_p + \delta\theta$ and with this value compute

$$\theta_2 = \theta_1 + \Delta\theta$$

$$e = \frac{\rho_2 - \rho_1}{\rho_1 \cos\theta_1 - \rho_2 \cos\theta_2}$$

$$p = \rho_1(1 + e\cos\theta_1)$$

$$a = \frac{p}{1 - e^2}$$

and the eccentric anomalies E_1 and E_2 from the $F(\alpha,\beta)$ function defined previously to give:

$$E_1 = F(e,\theta_1)$$

$$E_2 = F(e,\theta_2)$$

and the elliptic time

$$T_2 - T_1 = \frac{1}{2\pi}\sqrt{a^3}\left[E_2 - E_1 - e(\sin E_2 - \sin E_1)\right]$$

This value of $T_2 - T_1$ is compared with Δt and if still too small, another increment $\delta\theta$ is added to θ_1 and the process repeated. When $T_2 - T_1$ is larger than Δt, then the interval $\delta\theta$ is halved and subtracted from the last value of θ_1. This method of iteration is free from the singularities involved in methods using derivatives and is not significantly longer. The value of h used affects the time of iteration; however, an arbitrary value of $h = 3$ seems to produce sufficiently fast convergence.

APPENDIX B

Hyperbolic transfer orbits.- A value of $\Delta t < \Delta T_p$ gives hyperbolic transfer orbits and hence $-\psi < \theta_1 < \theta_p$. The method of iteration is identical to that for the elliptic case although the equations differ slightly. These equations appear as

$$\delta\theta = \frac{\psi + \theta_p}{h} = \frac{\theta_p - (-\psi)}{h}$$

$$\theta_1 = \theta_p - \delta\theta$$

$$\theta_2 = \theta_1 + \Delta\theta$$

The iteration begins at θ_p as in the elliptic case but now proceeds into the region of smaller θ_1

$$e = \frac{\rho_2 - \rho_1}{\rho_1 \cos \theta_1 - \rho_2 \cos \theta_2}$$

$$p = \rho_1(1 + e \cos \theta_1)$$

$$a = \frac{p}{1 - e^2} \qquad (a < 0)$$

The hyperbolic time functions must now be used and equation (B5) is applicable if γ is set equal to -1.0; thus, $\sqrt{\frac{\alpha - 1}{1 + \alpha}}$ is obtained. Proceed with the usual hyperbolic "eccentric anomalies" which are the $F(\alpha,\beta)$ functions with the preceding replacements.

$$F_1 = F'(e,\theta_1)$$

$$F_2 = F'(e,\theta_2)$$

and the times from $\theta = 0$ to $\theta = \theta_1$ and $\theta = \theta_2$ as

$$T_1 = \frac{1}{2\pi}\sqrt{-a^3}\left\{e \tan F_1 - ln\left[\tan\left(\frac{F_1}{2} + \frac{\pi}{4}\right)\right]\right\}$$

$$T_2 = \frac{1}{2\pi}\sqrt{-a^3}\left\{e \tan F_2 - ln\left[\tan\left(\frac{F_2}{2} + \frac{\pi}{4}\right)\right]\right\}$$

Compare $T_2 - T_1$ with Δt. If it is still larger, compute again with a new θ_1 decreased from the last by $\delta\theta$. If $T_2 - T_1$ is greater, increase θ_1 by $\frac{\delta\theta}{2}$ and proceed in this manner until the desired agreement is obtained.

Special cases.- In the case that $\rho_1 = \rho_2$ in either the elliptic or hyperbolic case, special computations are necessary. For instance, the iteration can no longer be accomplished by incrementing θ_1 as θ_1 becomes a fixed value $-\psi$. The problem is solved by iterating the eccentricity in a straightforward manner and is easily followed in the flow chart in appendix D.

In the case that the time ΔT_p happens to be equal to Δt, then there is no further iteration as the solution is the parabolic case.

Computation of the properties.- After obtaining θ_1, information on the rates of change is obtained. Also the quantities ρ_1 and ρ_2 must be reassociated with r_f and ξ_i. The properties of the transfer orbit occurring at $t = t_i$, $t = t_f$ are defined as ρ_i, ρ_f, θ_i, and θ_f where

$$\rho_i = \rho_1 \qquad (\xi_i < r_f)$$

$$\rho_i = \rho_2 \qquad (\xi_i > r_f)$$

$$\rho_f = \rho_2 \qquad (r_f > \xi_i)$$

$$\rho_f = \rho_1 \qquad (r_f < \xi_i)$$

$$\theta_i = \theta_1 \qquad (\xi_i < r_f)$$

$$\theta_i = -\theta_2 \qquad (\xi_i > r_f)$$

$$\theta_f = \theta_2 \qquad (r_f > \xi_i)$$

$$\theta_f = -\theta_1 \qquad (r_f < \xi_i)$$

These relations follow directly from the geometry of the transfer orbit and the definition that ρ_1 is the minimum of r_f and ξ_i and ρ_2 is the maximum of r_f and ξ_i. The rates of change at $t = t_i$; $t = t_f$ in the plane of the transfer orbit may now be computed. Immediately after the impulse at $t = t_i$,

$$\dot{\theta}_i = \sqrt{p}\,\frac{1}{\rho_i^2}$$

$$\dot{\rho}_i = \sqrt{\frac{1}{p}}\, e \sin \theta_i$$

and immediately before the impulse at $t = t_f$,

$$\dot{\theta}_f = \sqrt{p}\,\frac{1}{\rho_f^2}$$

$$\dot{\rho}_f = \sqrt{\frac{1}{p}}\, e \sin \theta_f$$

The transfer orbit plane is defined in exactly the same way as the initial interceptor orbit with the elements Ω, ω, i, and θ_1. The anomaly θ_1 and inclination i were found previously. Once Ω and ω have been determined, the transformation matrix $[b_{ij}]$ for transforming the position and velocity vectors in x', y', and z' (see fig. 3) to components in X,Y,Z is needed. This transformation matrix is identical with the $[a_{ij}]$ transformation matrix where ω_I, Ω_I, and i_I are replaced by ω, Ω, and i.

The velocity vector at $t = t_i$ in the x' direction is $\dot{\rho}_i(x')$, y' is $\dot{\rho}_i(y')$, z' is $\dot{\rho}_i(z')$ where

$$\left.\begin{aligned} \dot{\rho}_i(x') &= \dot{\rho}_i \cos \theta_i - \rho_i\dot{\theta}_i \sin \theta_i \\ \dot{\rho}_i(y') &= \dot{\rho}_i \sin \theta_i + \rho_i\dot{\theta}_i \cos \theta_i \\ \dot{\rho}_i(z') &= 0 \end{aligned}\right\} \tag{B42}$$

The velocity components at $t = t_f$ in the x',y',z' coordinate directions are

$$\left.\begin{aligned} \dot{\rho}_f(x') &= \dot{\rho}_f \cos \theta_f - \rho_f\dot{\theta}_f \sin \theta_f \\ \dot{\rho}_f(y') &= \dot{\rho}_f \sin \theta_f + \rho_f\dot{\theta}_f \cos \theta_f \\ \dot{\rho}_f(z') &= 0 \end{aligned}\right\} \tag{B43}$$

These velocity components may be transformed to components in the X,Y,Z coordinate system by use of the $[b_{ij}]$ matrix above.

From these results the velocities may be determined at $t = t_i$ in the initial interceptor orbit and $t = t_f$ in the target plane. Then at $t = t_i$ in the x,y,z coordinate system:

$$\left.\begin{aligned} \dot{\xi}_i(x) &= \dot{\xi}_i \cos \nu_i - \xi_i\dot{\nu}_i \sin \nu_i \\ \dot{\xi}_i(y) &= \dot{\xi}_i \sin \nu_i + \xi_i\dot{\nu}_i \cos \nu_i \\ \dot{\xi}_i(z) &= 0 \end{aligned}\right\} \tag{B44}$$

which may be transformed to X,Y,Z coordinates by the matrix $[a_{ij}]$.

Finally, obtain the velocity after the final impulse directly from the components X,Y,Z without transformation

$$\left.\begin{aligned} \dot{r}_f(X) &= \dot{r}_f \cos \phi_f - r_f\dot{\phi}_f \sin \phi_f \\ \dot{r}_f(Y) &= \dot{r}_f \sin \phi_f + r_f\dot{\phi}_f \cos \phi_f \\ \dot{r}_f(Z) &= 0 \end{aligned}\right\} \tag{B45}$$

The Velocities in X, Y, and Z

The matrix operations on these vectors give the desired velocity components in the Newtonian frame X,Y,Z. These components are denoted by $V_k(j)$ where the subscript k refers to the time (where k equals 1 and 2 just before and after the initial impulse and k equals 3 and 4 just before and after the final impulse) and j to the component X,Y,Z. These velocity components, in terms of the components shown in equations (B42), (B43), (B44), (B45), and the elements of the $[a_{ij}]$ and $[b_{ij}]$ matrices, are given in the flow diagram of appendix D.

The velocity increments required to perform the rendezvous maneuver are easily found by subtracting these components. For the velocity increment at $t = t_1$, it is necessary to subtract the corresponding components of state (1) from state (2) as

$$\begin{aligned} \Delta V_i(X) &= V_2(X) - V_1(X) \\ \Delta V_i(Y) &= V_2(Y) - V_1(Y) \\ \Delta V_i(Z) &= V_2(Z) - V_1(Z) \end{aligned}$$

and for the terminal maneuver subtract state (3) from state (4) to give the desired increments

$$\begin{aligned} \Delta V_f(X) &= V_4(X) - V_3(X) \\ \Delta V_f(Y) &= V_4(Y) - V_3(Y) \\ \Delta V_f(Z) &= V_4(Z) - V_3(Z) \end{aligned}$$

The total velocity increment for the maneuver is the sum of the maneuvers at $t = t_i \left(\Delta V_1\right)$ and $t = t_f \left(\Delta V_2\right)$

$$\Delta V_i = \sqrt{\Delta V_i(X)^2 + \Delta V_i(Y)^2 + \Delta V_i(Z)^2}$$

APPENDIX B

$$\Delta V_f = \sqrt{\Delta V_f(X)^2 + \Delta V_f(Y)^2 + \Delta V_f(Z)^2}$$

and

$$\Delta V = \Delta V_i + \Delta V_f$$

APPENDIX C

DESCRIPTION OF PROGRAM

This program was written in the FORTRAN IV (Ibsys version 9) language for the IBM 7094 computer at the Langley Research Center. Throughout the program there is an emphasis on simplicity, but capability has been provided for several uses and for a freedom of choice on input and output.

The method of solving the problem can readily be obtained by following the flow diagram (appendix D) and the description given in appendix B. The description in this appendix gives additional information about the options available, types of input and output, criteria for testing variables for transfer in the program, criteria for testing the convergence in the iterative processes, and criteria for incrementing the times. The methods used in the iterative processes are described fully in appendix B and can be followed on the flow diagram (appendix D). The flow diagram also shows the methods of incrementing Δt_w, Δt, and ν_o.

A complete listing of the FORTRAN IV program is given as appendix E.

Subprograms

Six small subprograms are used in addition to the main program. They are called: AAA, VELOC, SIX, ANOM, SPACE, and CONVRT. An explanation of the computations, provided by AAA and ANOM, are included in appendix B. Equations (B5) are contained in AAA and equations (B9) are contained in ANOM. The other subprograms are self-explanatory; however, a brief description of the uses of all six subprograms is given here.

AAA is used to compute the eccentric anomaly if the true anomaly is known or vice versa.

VELOC is used to compute the velocity components.

SIX is used to compute the elements of the $[a_{ij}]$ and $[b_{ij}]$ matrices.

ANOM is used to compute the eccentric anomaly if the mean anomaly is known.

SPACE is used to test the line count and to skip pages and print column headings when necessary.

CONVRT is used to convert the times from dimensionless quantities to units corresponding to input for printing purposes.

APPENDIX C

Options Available

The program offers several options for choosing the variable which will be incremented initially and for choosing the type of input. This choice is made by reading in three control factors: GUIDE, OPTION, and DATAS.

GUIDE determines whether Δt_w or Δt varies initially in the program and gives the appropriate output format regarding the choice. The program provides that the time Δt_w or Δt not incremented initially will be incremented secondly, after which ν_o will be incremented. This procedure works for any number of Δt, Δt_w, and ν_o values. The quantity incremented initially will appear as the abscissa for ease in plotting or analyzing the data. Hence,

if GUIDE = 1, Δt_w varies initially;
if GUIDE = 2, Δt varies initially.

OPTION provides a choice of three types of input for times:

OPTION = 1, Δt_w, Δt input dimensionless;
OPTION = 2, Δt_w, Δt input in minutes; and
OPTION = 3, Δt_w, Δt input in days.

Restrictions are placed on maximums for Δt_w and Δt because of the six spaces allowed by the output format for printout of the quantity varying initially. The restrictions are due only to the output format, and may easily be changed.

(a) If either time exceeds 9999.0 minutes, they must be read in either dimensionless or in days.

(b) If either exceeds 9999.0 days, they must be read in dimensionless.

(c) If either exceeds 9.9999 in dimensionless time and is restricted by (a) or (b), the leftmost characters of the value will not be printed on output.

DATAS provide a choice of input:

if DATAS = 1, use Keplerian input;
if DATAS = 2, use relative orbital input.

Input

Information on input may be found in the "Computer Input and Output" section of the paper and in the immediately preceding paragraphs. Input is to be made in units according to the following criteria:

(1) The quantities μ and a_T are dimensioned, unless time is dimensionless in which case they may be in any units desired. However, in order to get values for P_T and V_{CT}, μ and a_T should be dimensioned.

(2) All angles are in degrees.

(3) Times are in units described under "Options Available" in this appendix.

(4) All other quantities are dimensionless.

Input cards in the order to be read in and the proper FORTRAN formats for each are listed in the following table:

Order of cards	Variable names	FORTRAN format
1	GUIDE, ØPTIØN, DATAS	4E18.8
2	AMU, AT	4E18.8
3	AH, RMIN, ET, PHIO	4E18.8
4	DTWI, DTWF, DTTI, DTTF	4E18.8
5	DELTW, DELTT	4E18.8
6	Keplerian input:	4E18.8
	EI, AI, ØMEGAI, ØMI	
	Relative orbital input:	4E18.8
	XO, YO, ZO	
7	Keplerian input:	4E18.8
	AII, ANUO1, ANUOL, DELNUO	
	Relative orbital input:	
	XODT, YODT, ZODT	

Output

All output will be in units corresponding to input. The output format will vary as Δt and Δt_W are varied initially. When a set of data is read in, those initial conditions are printed. As ν_O and Δt or Δt_W are incremented, they are printed. The computed values are then printed in columns. Special notation and messages which may be printed are

(1) If relative orbital input is used and a_I is computed as less than zero, a message to this effect will be written and the program will transfer to the initial input section.

(2) If $i \geqq 90^{O} - 0.1^{O}$, the message "i = +/-90 degrees not acceptable data" is written.

(3) If the orbit is parabolic, the value for a will appear as 99.999 instead of ∞.

(4) If $\rho_{min} < r_{min}$, an asterisk is printed at the extreme right-hand end of the line of data.

Testing Criteria

There are several places in the program where tolerance factors or allowances for computational inaccuracies must be defined to insure proper flow through the program. An explanation of the values chosen in these tests follows:

Values compared	Remarks
$\frac{H_{o3}}{H_o}$ with 0.99999995	This test compares the value of the cos i_I, $\left(\frac{H_{o3}}{H_o}\right)$, to 1.0, and restricts the minimum value of a computed i_I to be 0.0001 radian; otherwise, i_I is set equal to 0
$\|i\|$ with $90^o - 0.1^o$	Data which result in an i of 90^o are not acceptable in this program. The absolute value of i is tested against 90^o with a margin of 0.1^o
ΔT_p with Δt	This test of the parabolic time against the transfer time to determine the type of orbit results in a parabolic orbit only if the values are equal. A tolerance of $0.5 \times 10^{-6}(\Delta t)$ defined as CRIT was allowed at the point of equivalence for computational error
i with 0.5×10^{-7}	For this test a margin of tolerance of 0.5×10^{-7} was allowed for machine inaccuracy
ΔT with Δt	This test is made to determine when the convergence is sufficient to leave the loop in the iteration process. A value CRIT = $0.5 \times 10^{-6}(\Delta t)$ was defined to give a tolerance margin based on the value of the transfer time and to cover any computational error
$\delta\theta$ with 1.0×10^{-7} δe with 1.0×10^{-7}	The appropriate test is made in each iteration scheme to test for the effectiveness of the increment on the variable. If the increment is equal to or less than the value tested against, its effect on the variable is negligible and the iteration is ended. This procedure acts as a safety check for ending the iteration in the event that ΔT never gets within the prescribed range (CRIT) of Δt
ρ_1 with ρ_2	A tolerance margin of $1.0 \times 10^{-6}(a_I)$, defined as CRIRHØ, provides for transfer to the special iteration necessary when $\rho_1 = \rho_2$
$\Delta\theta$ with 0.0001^o $\Delta\theta$ with $360^o - 0.0001^o$	Restrictions are put on a margin of 0.0001^o around $\Delta\theta = 0^o(360^o)$ because of computational sensitivity. For these cases, transfer orbit properties are set equal to the initial target orbit properties and the iterations are omitted

APPENDIX C

The quantities used for testing in incrementing Δt_w, Δt, and ν_o are as follows:

$$\text{TWFMDT} = \Delta t_w(\text{last}) - 1.5(\delta t_w) - \text{TESTTW}$$

$$\text{TTFMDT} = \Delta t(\text{last}) - 1.5(\delta t) - \text{TESTTT}$$

$$\text{ANLMDN} = \nu_o(\text{last}) - 1.5(\delta \nu_o) - 1.0 \times 10^{-7}$$

$$\text{DTWFT} = \Delta t_w(\text{last}) - \text{TESTTW}$$

$$\text{DTTFT} = \Delta t(\text{last}) - \text{TESTTT}$$

$$\text{ANUOLT} = \nu_o(\text{last}) - 1.0 \times 10^{-7}$$

$$\text{TESTTW} = 0.5 \times 10^{-6}\left[\Delta t_w(\text{last})\right], \text{ or } = 1.0 \times 10^{-7}$$

$$\text{TESTTT} = 0.5 \times 10^{-6}\left[\Delta t(\text{last})\right], \text{ or } = 1.0 \times 10^{-7}$$

The TWFMDT is used to test against Δt_w and allows it to be incremented by δt_w until Δt_w is greater than or equal to $\Delta t_w(\text{last}) - 1.5\ \delta t_w$. A value TESTTW is arbitrarily chosen as 1.0×10^{-7} or $0.5 \times 10^{-6}\left[\Delta t_w(\text{last})\right]$ depending upon whether δt_w is 0 or is greater than 0, respectively. The TESTTW value is included to provide a tolerance for computational error in the value of TWFMDT if the δt_w is equal to 0 and to insure that control is transferred out of this loop after computation with $\Delta t_w(\text{first})$. The DTWFT is used to test against Δt_w and always provides the computation for $\Delta t_w(\text{last})$, even if $\dfrac{\Delta t_w(\text{last}) - \Delta t_w(\text{first})}{\delta t_w}$ is not an integer. This separate control for the final point also insures ease in incorporating any additional desired statements or controls, such as those for plotting routines, in the program at the end of a series of incrementations. The TESTTW in the expression for DTWFT provides tolerance for machine error.

The same explanation applies for TTFMDT, DTTFT, and TESTTT, if these variable names replace TWFMDT, DTWFT, and TESTTW in the preceding paragraph. These variables are used for testing in the incrementing procedure for Δt.

A similar explanation for ANLMDN and ANUOLT, used for incrementing ν_o, holds with the exception that the tolerance margin or computational error is chosen as 1.0×10^{-7} for both values, instead of being a ratio as in TESTTW and TESTTT, because ν_o is in radians (in the form X.XXXXXXX) and the 1.0×10^{-7} will always be effective on the number of places used in the computation.

APPENDIX C

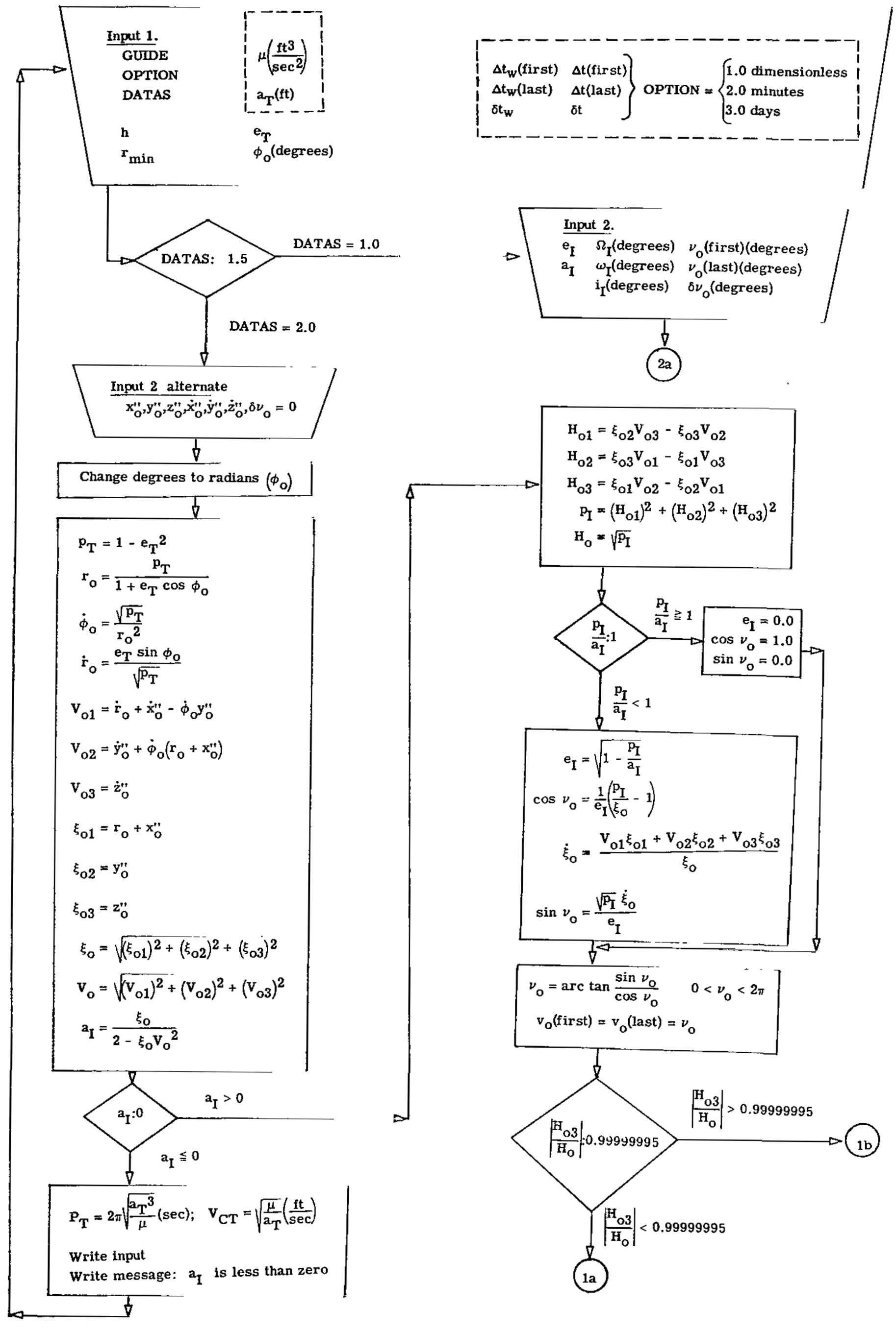
The mathematical symbol and its FORTRAN equivalent are given in the following table:

Mathematical symbol	Fortran symbol
a	AT,AI,A
a_{ij}	A11,A12,A21,A22,A31,A32
b_{ij}	B11,B12,B21,B22,B31,B32
c	C
e	ET,EI,E
E	ETO,ETF,EIO,EII,EL,E2
F	F1,F2
h	AH
H	AHO1,AHO2,AHO3,AHO
i	BI,AII,I,II
M	AMTO,AMTF,AMIO,AMII,AM1
p	PI,PT,P,PP
P_T	TPER
r	RO,RODT,RF,RDTF,RDTF1,RDTF2,RMIN
Δt_w	DTWI,DTWF,DELTTW
Δt	DTTI,DTTF,DELTTT
T_1,T_2	T1,T2,TP1,TP2
ΔT	DT,DELTTP,DELTTC
V	VO1,VO2,VO3,VO
V_{CT}	VCT
$V_k(j)$	V11,V12,V13,V21,V22,V23,V31,V32,V33,V41,V42,V43
$\Delta V_i(j),\Delta V_f(j)$	DV11,DV12,DV13,DV21,DV22,DV23
$\Delta V_i,\Delta V_f$	DELV1,DELV2
ΔV	DELV
$x'',y'',z'',\dot{x}'',\dot{y}'',\dot{z}''$	XO,YO,ZO,YODT,ZODT
$\bar{\alpha}$	ALPHA
γ	GAMMA
$\bar{\gamma}$	GAMBAR
$\Delta\theta$	DELTTH
θ	TH1,TH2,THP,THI,THF,THDTI,THDTF,THPMAX
$\delta\theta$	DELTH
μ	AMU,MU
ν	ANUO1,ANUOL,ANUO,ANUI,ANUDTI,NUO1,NUOL,NUO,NUI
ξ	XIO1,XIO2,XIO3,XIO,XIODT,XII,XIDTI,XII1,XII2,XII3,XIIDT1, XIIDT2
ρ	RHØ1,RHØ2,RHØI,RHØF,RHØDTI,RHØDTF,RØDTI1,RØDTI2, RØDTF1,RØDTF2,RHØM
ϕ	PHIO,PHIODT,PHIF,PHIDTF
ψ	PSI
ω	ØMI,ØM
Ω	ØMEGAI,ØMEGA

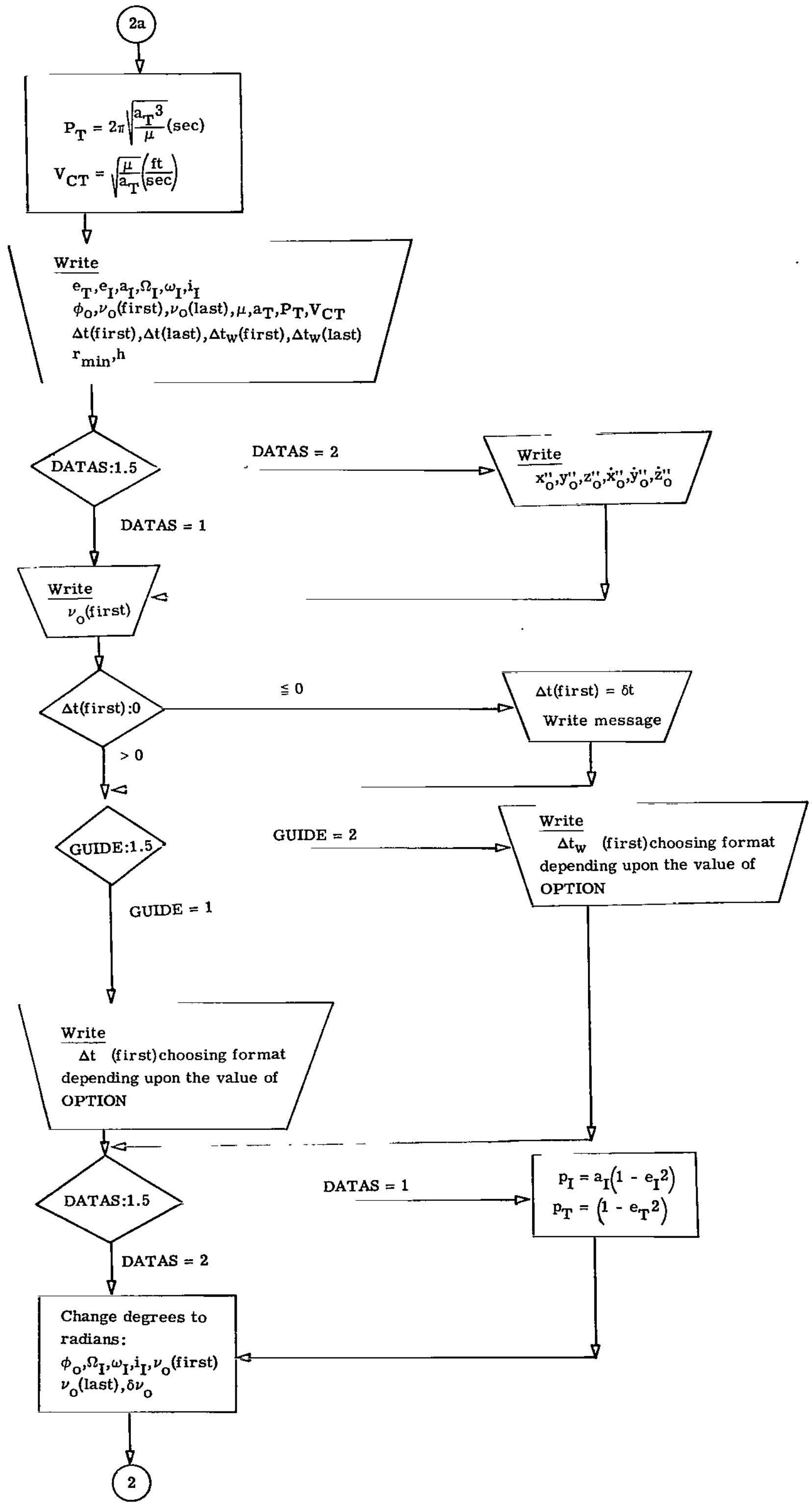
APPENDIX D

COMPUTATIONAL FLOW DIAGRAM

The computational flow diagram is given in this appendix. All quantities are dimensionless unless otherwise specified. Provisions for control of spacing on output are not included on this diagram.



APPENDIX D


2a
$P_T = 2\pi\sqrt{\frac{a_T^3}{\mu}}$ (sec)
$V_{CT} = \sqrt{\frac{\mu}{a_T}}\left(\frac{ft}{sec}\right)$
Write
$e_T, e_I, a_I, \Omega_I, \omega_I, i_I$
ϕ_o, ν_o(first), ν_o(last), μ, a_T, P_T, V_{CT}
Δt(first), Δt(last), Δt_w(first), Δt_w(last)
r_{min}, h
DATAS:1.5
DATAS = 2
Write
$x_o'', y_o'', z_o'', \dot{x}_o'', \dot{y}_o'', \dot{z}_o''$
DATAS = 1
Write
ν_o(first)
Δt(first):0
≤ 0
Δt(first) = δt
Write message
> 0
GUIDE:1.5
GUIDE = 2
Write
Δt_w (first) choosing format depending upon the value of OPTION
GUIDE = 1
Write
Δt (first) choosing format depending upon the value of OPTION
DATAS:1.5
DATAS = 1
$p_I = a_I(1 - e_I^2)$
$p_T = (1 - e_T^2)$
DATAS = 2
Change degrees to radians:
$\phi_o, \Omega_I, \omega_I, i_I, \nu_o$(first)
ν_o(last), $\delta\nu_o$
2

APPENDIX D

(1a)

$$i_I = \text{arc cos}\,\frac{H_{o3}}{H_o} \qquad 0 < i_I \leqq \pi$$

$$\cos\Omega_I = \frac{-H_{o2}\cos\phi_o + H_{o1}\sin\phi_o}{H_o\sin i_I}$$

$$\sin\Omega_I = \frac{H_{o1}\cos\phi_o - H_{o2}\sin\phi_o}{H_o\sin i_I}$$

$$\Omega_I = \text{arc tan}\,\frac{\sin\Omega_I}{\cos\Omega_I} \qquad 0 \leqq \Omega_I \leqq 2\pi$$

$$\cos(\omega_I + \nu_o) = \frac{\xi_{o2}H_{o1} - \xi_{o1}H_{o2}}{\xi_o H_o \sin i_I}$$

$$\sin(\omega_I + \nu_o) = \frac{\xi_{o3}}{\xi_o \sin i_I}$$

(1b)

$$i_I = 0$$

$$\Omega_I = 0$$

$$\cos(\omega_I + \nu_o) = \frac{\xi_{o1}\cos\phi_o - \xi_{o2}\sin\phi_o}{\xi_o}$$

$$\sin(\omega_I + \nu_o) = \frac{\xi_{o1}\sin\phi_o + \xi_{o2}\cos\phi_o}{\xi_o}$$

$$(\omega_I + \nu_o) = \text{arc tan}\,\frac{\sin(\omega_I + \nu_o)}{\cos(\omega_I + \nu_o)} \qquad 0 < (\omega_I + \nu_o) < 2\pi$$

$$\omega_I = (\omega_I + \nu_o) - \nu_o$$

Convert radians to degrees for printing ($\phi_o, \Omega_I, \omega_I, i_I, \nu_o$(first), ν_o(last), $\delta\nu_o$)

(2a)

(2)

Change Δt and Δt_w to dimensionless quantities:

$$\text{Option} = \begin{cases} 1.0 & \Delta t = \Delta t \\ 2.0 & \Delta t = 60\,\dfrac{\Delta t}{P_T} \\ 3.0 & \Delta t = 86400\,\dfrac{\Delta t}{P_T} \end{cases}$$

Compute quantities to use in testing for the completion of incrementing Δt_w, Δt, and ν_o: TESTTW, TESTTT, TWFMDT, TTFMDT, ANLMDN, DTWFT, DTTFT, ANUOLT

GUIDE: 1.5

GUIDE = 1

GUIDE = 2

$\nu_o = \nu_o$(first)

(2f)

$\Delta t_w = \Delta t_w$(first)

(2g)

$\Delta t = \Delta t$(first)

$\nu_o = \nu_o$(first)

(2c)

$\Delta t = \Delta t$(first)

(2d)

$\Delta t_w = \Delta t_w$(first)

(2h)

$$CRIT = 0.5 \times 10^{-6}(\Delta t)$$

CRIRHØ = 1.0E - 06* AI

$$\gamma = 1.0$$

* $E_{To} = S_1(e_T, \phi_o, \gamma)$

$$M_{To} = E_{To} - e_T \sin E_{To}$$

$$M_{Tf} = M_{To} + 2\pi(\Delta t + \Delta t_w)$$

** $E_{Tf} = S_2(e_T, M_{Tf})$

$$\phi_f = S_1(-e_T, E_{Tf}, \gamma)$$

$$r_f = \frac{P_T}{1 + e_T\cos\phi_f}$$

$$\dot{\phi}_f = \frac{\sqrt{P_T}}{r_f^2}$$

$$\dot{r}_f = \frac{e_T\sin\phi_f}{\sqrt{P_T}}$$

$$E_{Io} = S_1(e_I, \nu_o, \gamma)$$

$$M_{Io} = E_{Io} - e_I \sin E_{Io}$$

$$M_{Ii} = M_{Io} + \frac{2\pi(\Delta t_w)}{\sqrt{a_I^3}}$$

$$E_{Ii} = S_2(e_I, M_{Ii})$$

$$\nu_i = S_1(-e_I, E_{Ii}, \gamma)$$

$$\xi_i = \frac{P_I}{1 + e_I\cos\nu_i}$$

$$\dot{\nu}_i = \frac{\sqrt{P_I}}{\xi_i^2}$$

$$\dot{\xi}_i = \frac{e_I\sin\nu_i}{\sqrt{P_I}}$$

(3)

*S_1 in this diagram refers to Function Subprogram AAA.

**S_2 in this diagram refers to Subroutine Subprogram ANOM.

APPENDIX D

3

Determination of $\Delta\theta$, i of transfer orbit

$$\xi_{i1} = \xi_i\left[\cos \Omega_I \cos(\nu_i + \omega_I) - \cos i_I \sin \Omega_I \sin(\nu_i + \omega_I)\right]$$

$$\xi_{i2} = \xi_i\left[\sin \Omega_I \cos(\nu_i + \omega_I) + \cos i_I \cos \Omega_I \sin(\nu_i + \omega_I)\right]$$

$$\xi_{i3} = \xi_i\left[\sin i_I \sin(\nu_i + \omega_I)\right]$$

$$i = \arc\tan \frac{\xi_{i3}}{\xi_{i1} \sin \phi_f - \xi_{i2} \cos \phi_f} \qquad -\frac{\pi}{2} < i < \frac{\pi}{2}$$

$|i| : (90^\circ - 0.1^\circ)$

$|i| \geqq (90^\circ - 0.1^\circ)$: Write: $i = \pm 90^\circ$ not acceptable data → 11

$|i| < (90^\circ - 0.1^\circ)$:

$$\cos \Delta\theta = \frac{\xi_{i1} \cos \phi_f + \xi_{i2} \sin \phi_f}{\xi_i}$$

$$\sin \Delta\theta = \frac{\xi_{i1} \sin \phi_f - \xi_{i2} \cos \phi_f}{\xi_i \cos i}$$

$$\Delta\theta = \arc\tan \frac{\sin \Delta\theta}{\cos \Delta\theta} \qquad 0 \leqq \theta \leqq 2\pi$$

Computation of parabolic time

$\rho_2 = \max(r_f, \xi_i)$; $\rho_1 = \min(r_f, \xi_i)$

if $r_f = \xi_i$, then $\rho_1 = \rho_2$

$\Delta\theta : 0.001^\circ$

$\Delta\theta \leqq 0.001^\circ$: Write message; $a = 1.0$; $p = p_T$; $\theta_1 = \phi_f - \Delta\theta$; $\theta_2 = \phi_f$; $e = e_T$ → 8

$\Delta\theta > 0.001^\circ$: $\Delta\theta : 360^\circ - 0.001^\circ$

$\Delta\theta \geqq 360^\circ - 0.001^\circ$: → Write message ($a = 1.0$, ...) → 8

$\Delta\theta < 360^\circ - 0.001^\circ$:

$$c = \sqrt{\rho_1^2 + \rho_2^2 - 2\rho_1\rho_2 \cos \Delta\theta}$$

$$\sin \psi = \frac{\rho_1 - \rho_2 \cos \Delta\theta}{c}$$

$$\cos \psi = \frac{\rho_2 \sin \Delta\theta}{c}$$

$$\psi = \arc\tan \frac{\sin \psi}{\cos \psi} \qquad -\frac{\pi}{2} \leqq \psi \leqq \frac{3\pi}{2}$$

$$\bar{\alpha} = \sin^{-1} \frac{\rho_2 - \rho_1}{c} \qquad 0 \leqq \bar{\alpha} \leqq \frac{\pi}{2}$$

$$\theta_p = \bar{\alpha} - \psi \qquad \text{(Note: use of } \theta_p = (\theta_p)_{min}\text{)}$$

$$p_p = \rho_1(1 + \cos \theta_p)$$

$$T_{p1} = \frac{1}{2\pi}\sqrt{p_p^3}\left[\frac{1}{6}\tan^3\left(\frac{\theta_p}{2}\right) + \frac{1}{2}\tan\left(\frac{\theta_p}{2}\right)\right]$$

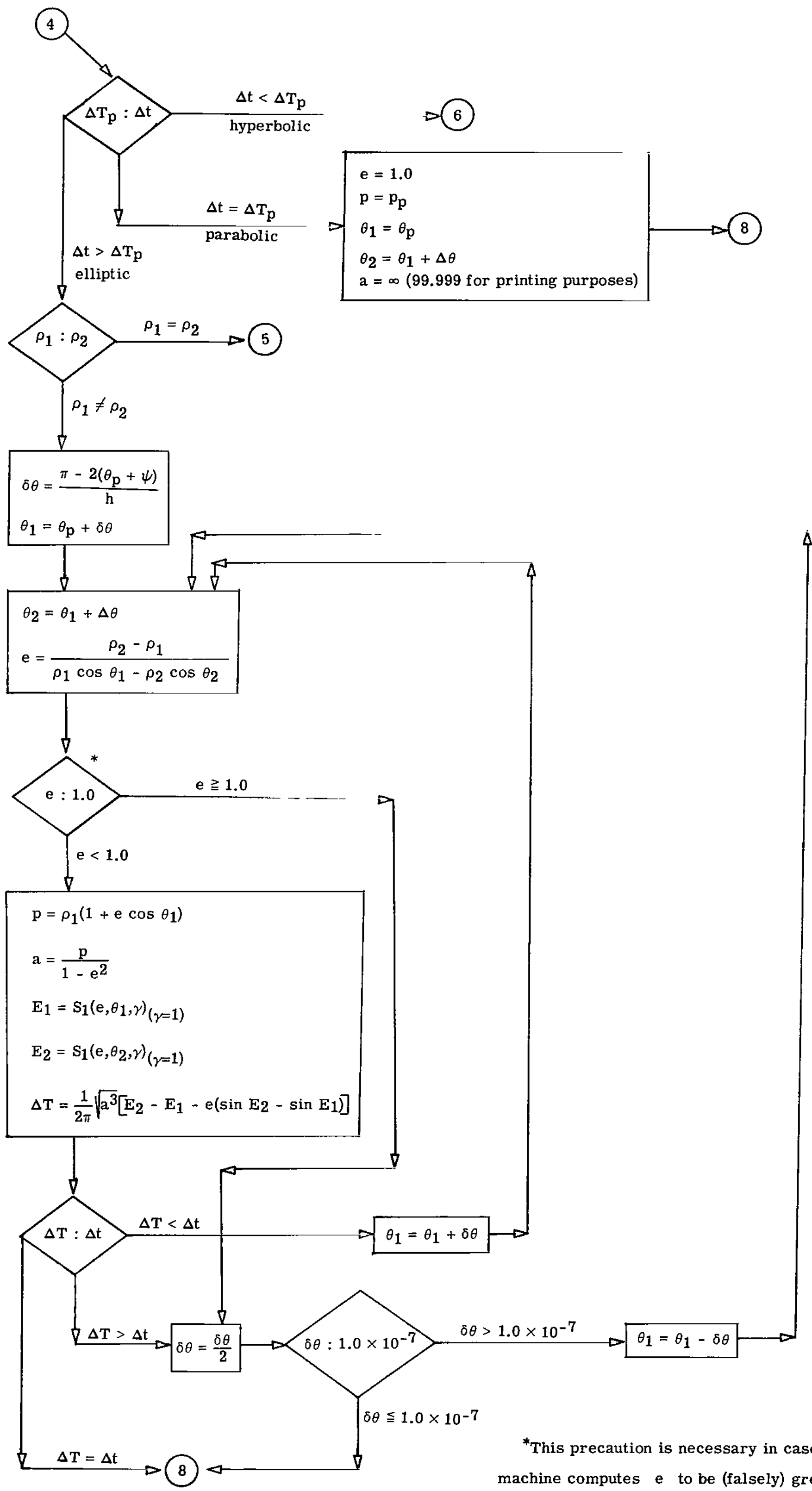
$$T_{p2} = \frac{1}{2\pi}\sqrt{p_p^3}\left[\frac{1}{6}\tan^3\left(\frac{\theta_p + \Delta\theta}{2}\right) + \frac{1}{2}\tan\left(\frac{\theta_p + \Delta\theta}{2}\right)\right]$$

$$\Delta T_p = T_{p2} - T_{p1}$$

$$(\theta_p)_{max} = \pi - (\alpha + \psi)$$

→ 4

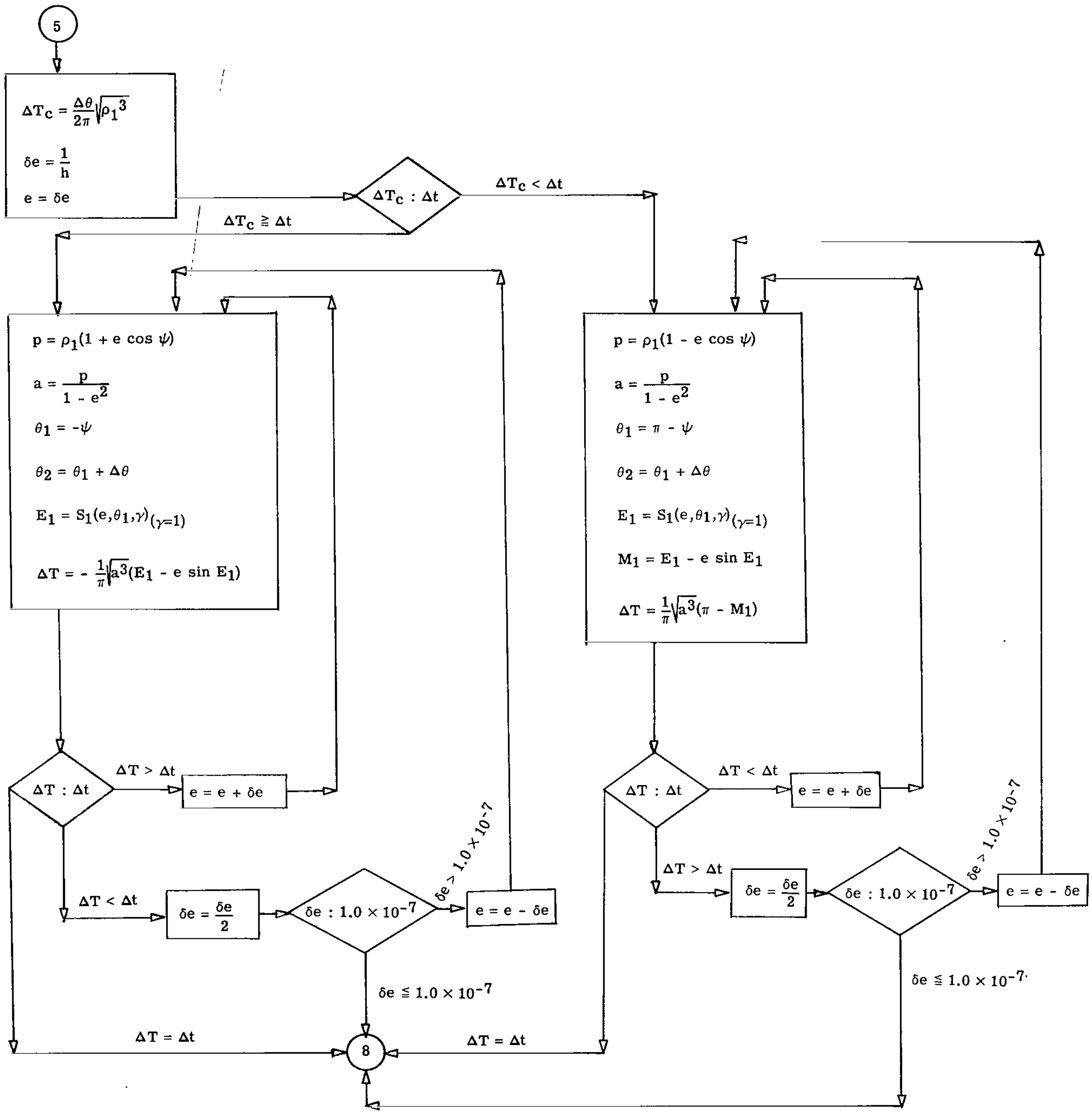
APPENDIX D



*This precaution is necessary in cases where $\theta_1 \approx \theta_{p_{max}}$ and machine computes e to be (falsely) greater than 1.

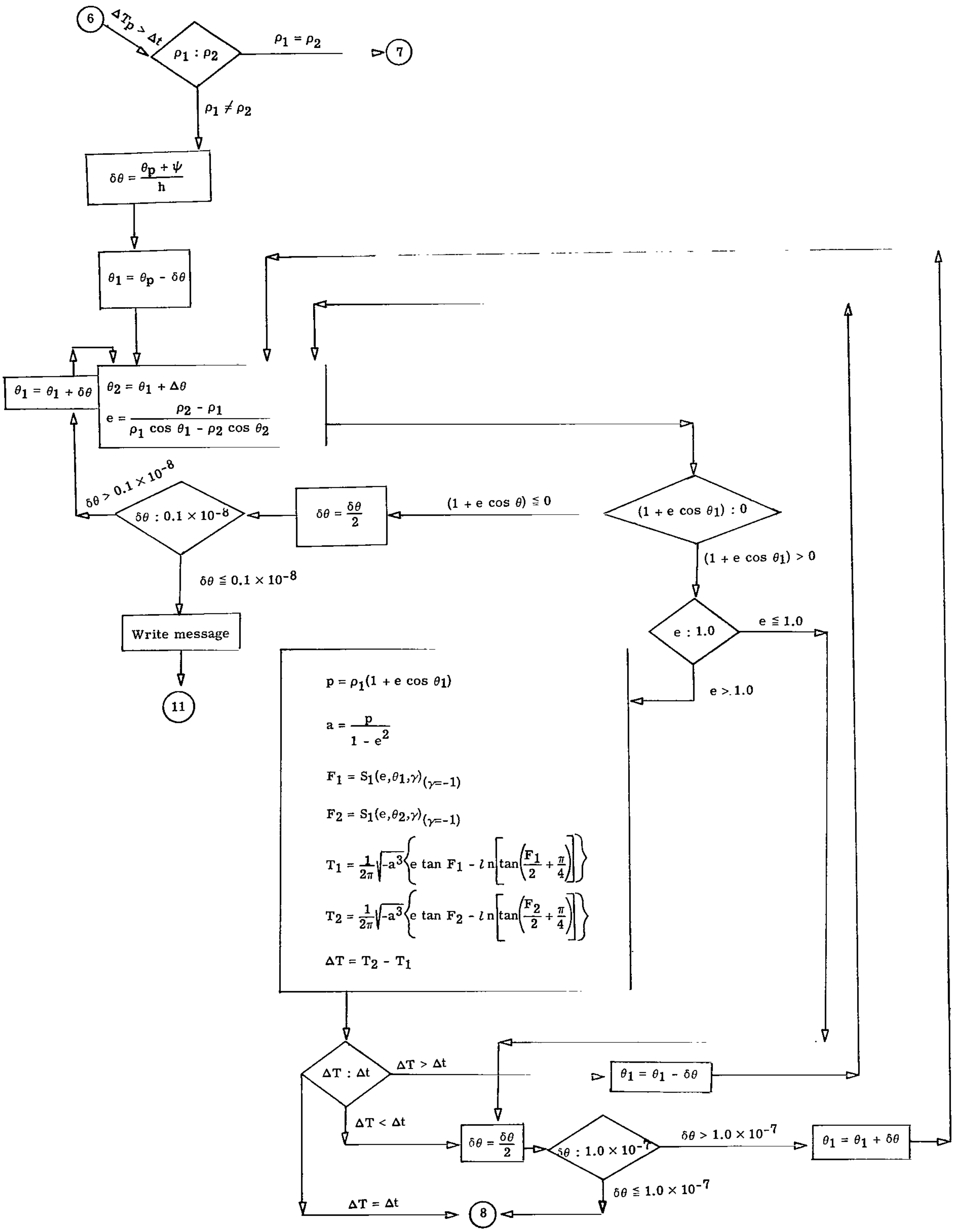
APPENDIX D

elliptic ($\rho_1 = \rho_2$)

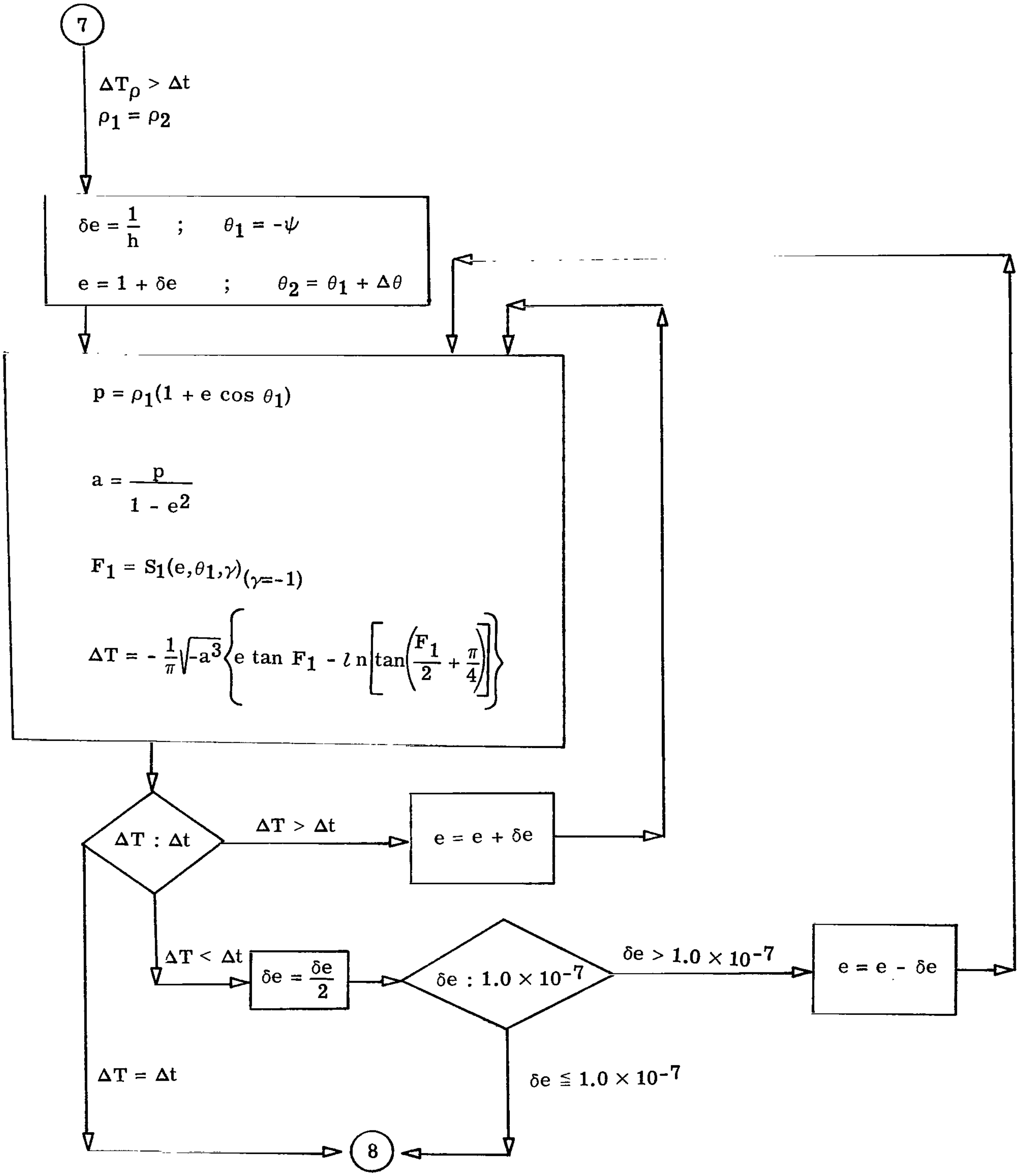


APPENDIX D

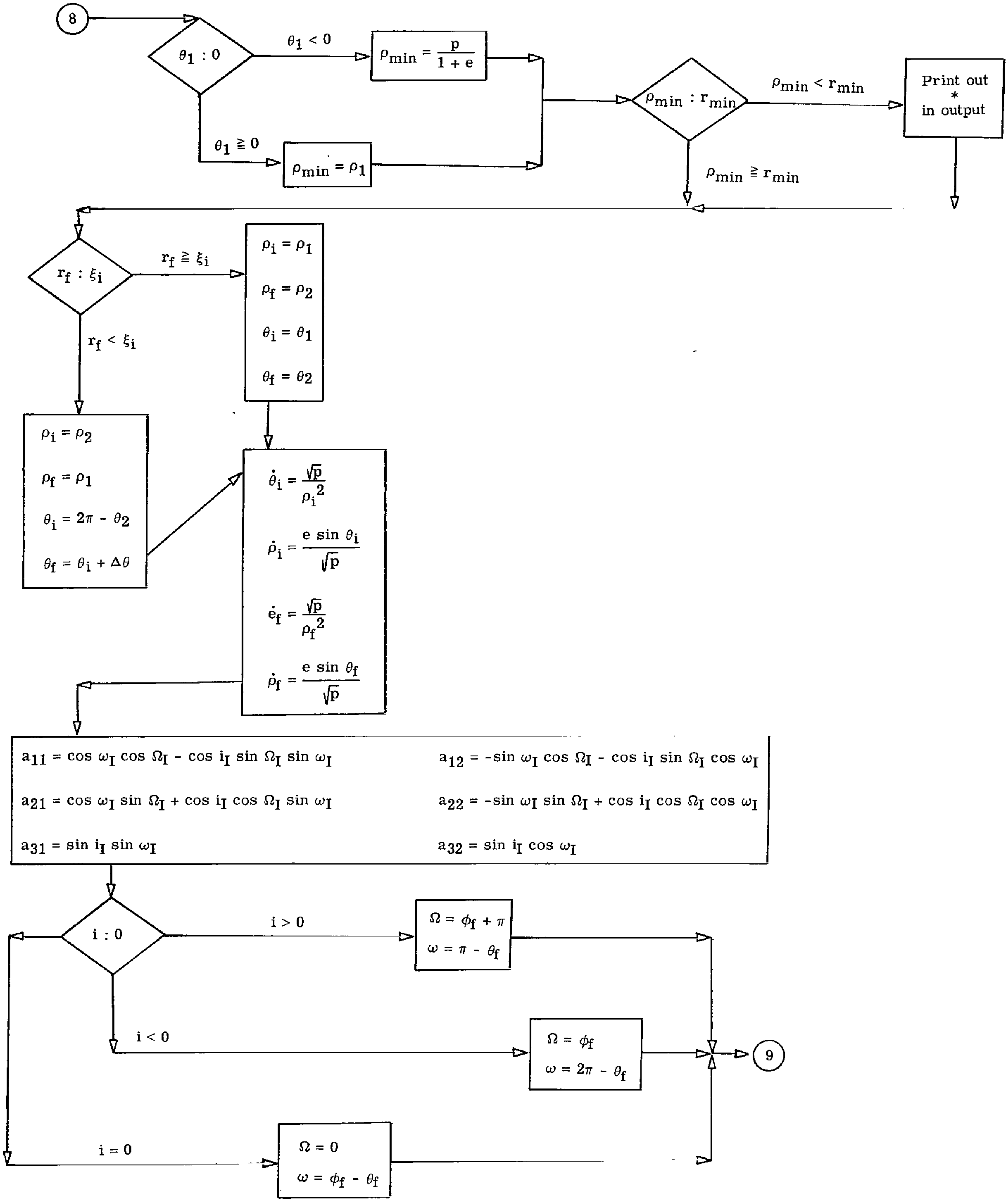
hyperbolic loop $\rho_1 \neq \rho_2$



hyperbolic loop ($\rho_1 = \rho_2$)



APPENDIX D



8
$\theta_1 : 0$
$\theta_1 < 0$
$\rho_{min} = \frac{p}{1+e}$
$\theta_1 \geqq 0$
$\rho_{min} = \rho_1$
$\rho_{min} : r_{min}$
$\rho_{min} < r_{min}$
Print out * in output
$\rho_{min} \geqq r_{min}$
$r_f : \xi_i$
$r_f \geqq \xi_i$
$\rho_i = \rho_1$
$\rho_f = \rho_2$
$\theta_i = \theta_1$
$\theta_f = \theta_2$
$r_f < \xi_i$
$\rho_i = \rho_2$
$\rho_f = \rho_1$
$\theta_i = 2\pi - \theta_2$
$\theta_f = \theta_i + \Delta\theta$
$\dot{\theta}_i = \frac{\sqrt{p}}{\rho_i^2}$
$\dot{\rho}_i = \frac{e \sin \theta_i}{\sqrt{p}}$
$\dot{\theta}_f = \frac{\sqrt{p}}{\rho_f^2}$
$\dot{\rho}_f = \frac{e \sin \theta_f}{\sqrt{p}}$
$a_{11} = \cos \omega_I \cos \Omega_I - \cos i_I \sin \Omega_I \sin \omega_I$
$a_{12} = -\sin \omega_I \cos \Omega_I - \cos i_I \sin \Omega_I \cos \omega_I$
$a_{21} = \cos \omega_I \sin \Omega_I + \cos i_I \cos \Omega_I \sin \omega_I$
$a_{22} = -\sin \omega_I \sin \Omega_I + \cos i_I \cos \Omega_I \cos \omega_I$
$a_{31} = \sin i_I \sin \omega_I$
$a_{32} = \sin i_I \cos \omega_I$
$i : 0$
$i > 0$
$\Omega = \phi_f + \pi$
$\omega = \pi - \theta_f$
$i < 0$
$\Omega = \phi_f$
$\omega = 2\pi - \theta_f$
$i = 0$
$\Omega = 0$
$\omega = \phi_f - \theta_f$
9

9

Matrix $\left[b_{ij}\right]$ same as $\left[a_{ij}\right]$

with Ω replacing Ω_I

ω replacing ω_I

i replacing i_I

$$\dot{\xi}_i(x) = \dot{\xi}_i \cos \nu_i - \xi_i \dot{\nu}_i \sin \nu_i$$

$$\dot{\xi}_i(y) = \dot{\xi}_i \sin \nu_i + \xi_i \dot{\nu}_i \cos \nu_i$$

$$\dot{\rho}_i(x') = \dot{\rho}_i \cos \theta_i - \rho_i \dot{\theta}_i \sin \theta_i$$

$$\dot{\rho}_i(y') = \dot{\rho}_i \sin \theta_i + \rho_i \dot{\theta}_i \cos \theta_i$$

$$\dot{\rho}_f(x') = \dot{\rho}_f \cos \theta_f - \rho_f \dot{\theta}_f \sin \theta_f$$

$$\dot{\rho}_f(y') = \dot{\rho}_f \sin \theta_f + \rho_f \dot{\theta}_f \cos \theta_f$$

$$\dot{r}_f(X) = \dot{r}_f \cos \phi_f - r_f \dot{\phi}_f \sin \phi_f$$

$$\dot{r}_f(Y) = \dot{r}_f \sin \phi_f + r_f \dot{\phi}_f \cos \phi_f$$

$$V_1(X) = a_{11} \dot{\xi}_i(x) + a_{12} \dot{\xi}_i(y)$$

$$V_1(Y) = a_{21} \dot{\xi}_i(x) + a_{22} \dot{\xi}_i(y)$$

$$V_1(Z) = a_{31} \dot{\xi}_i(x) + a_{32} \dot{\xi}_i(y)$$

$$V_2(X) = b_{11} \dot{\rho}_i(x') + b_{12} \dot{\rho}_i(y')$$

$$V_2(Y) = b_{21} \dot{\rho}_i(x') + b_{22} \dot{\rho}_i(y')$$

$$V_2(Z) = b_{31} \dot{\rho}_i(x') + b_{32} \dot{\rho}_i(y')$$

$$V_3(X) = b_{11} \dot{\rho}_f(x') + b_{12} \dot{\rho}_f(y')$$

$$V_3(Y) = b_{21} \dot{\rho}_f(x') + b_{22} \dot{\rho}_f(y')$$

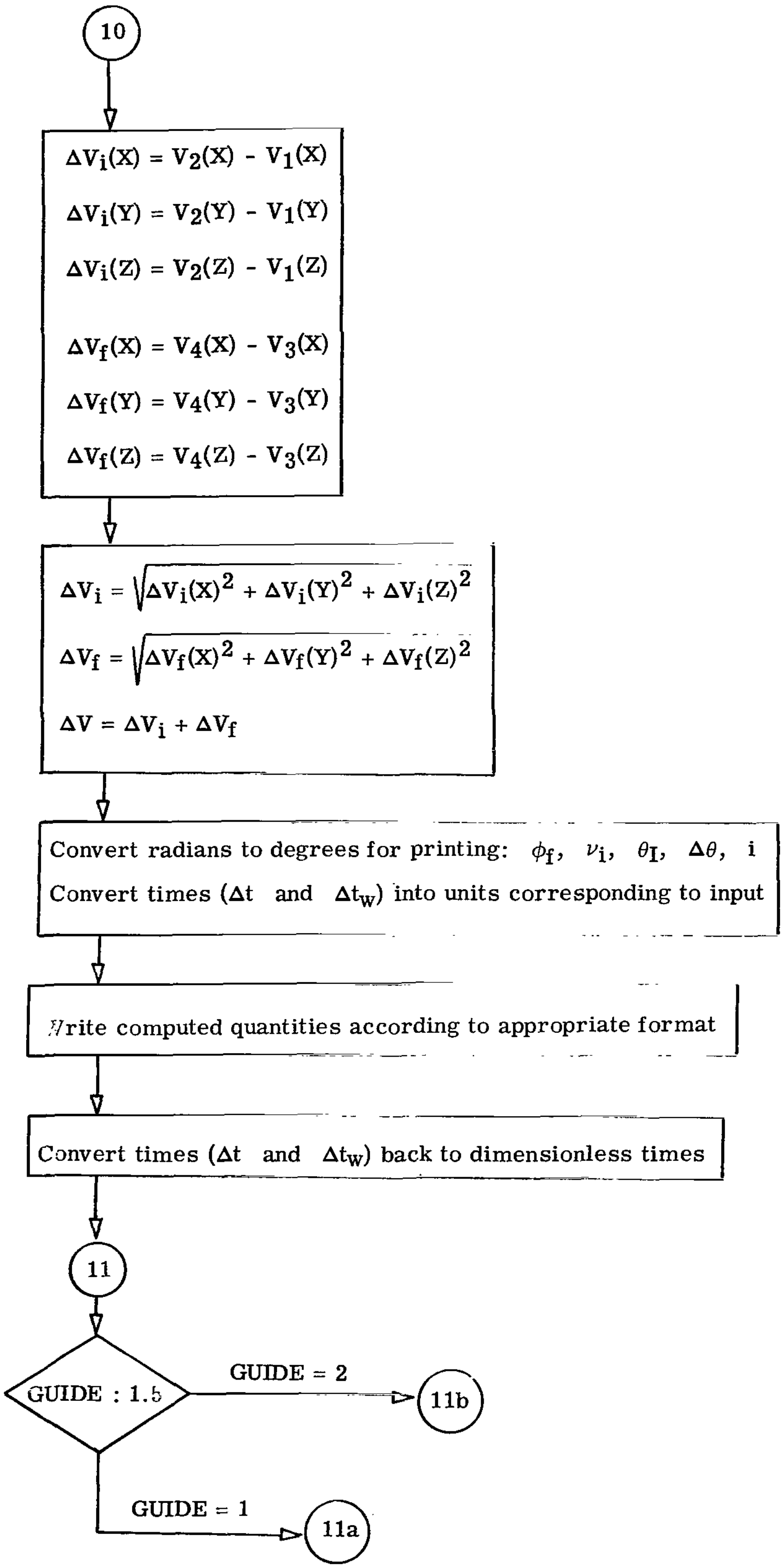
$$V_3(Z) = b_{31} \dot{\rho}_f(x') + b_{32} \dot{\rho}_f(y')$$

$$V_4(X) = \dot{r}_f(X)$$

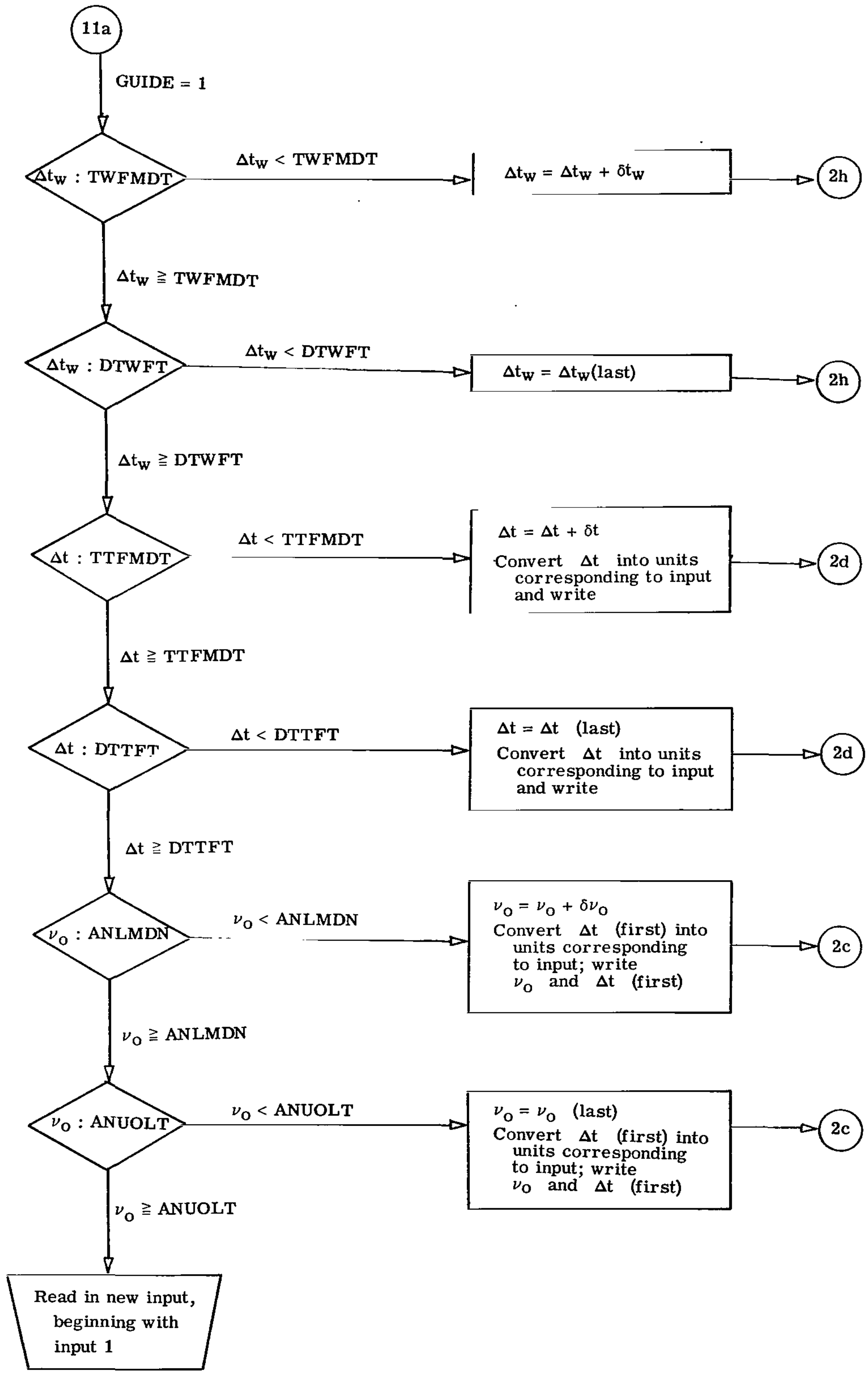
$$V_4(Y) = \dot{r}_f(Y)$$

$$V_4(Z) = 0$$

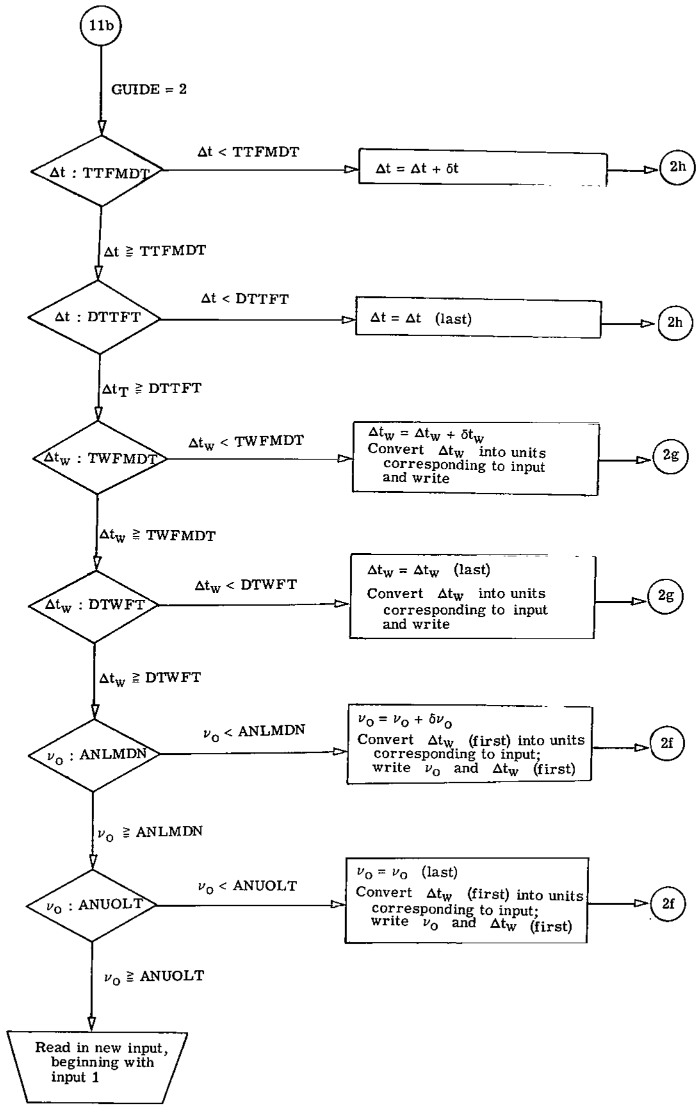
10


10
$\Delta V_i(X) = V_2(X) - V_1(X)$
$\Delta V_i(Y) = V_2(Y) - V_1(Y)$
$\Delta V_i(Z) = V_2(Z) - V_1(Z)$
$\Delta V_f(X) = V_4(X) - V_3(X)$
$\Delta V_f(Y) = V_4(Y) - V_3(Y)$
$\Delta V_f(Z) = V_4(Z) - V_3(Z)$
$\Delta V_i = \sqrt{\Delta V_i(X)^2 + \Delta V_i(Y)^2 + \Delta V_i(Z)^2}$
$\Delta V_f = \sqrt{\Delta V_f(X)^2 + \Delta V_f(Y)^2 + \Delta V_f(Z)^2}$
$\Delta V = \Delta V_i + \Delta V_f$
Convert radians to degrees for printing: ϕ_f, ν_i, θ_I, $\Delta\theta$, i
Convert times (Δt and Δt_w) into units corresponding to input
Write computed quantities according to appropriate format
Convert times (Δt and Δt_w) back to dimensionless times
11
GUIDE : 1.5
GUIDE = 2
11b
GUIDE = 1
11a

APPENDIX D



11a
GUIDE = 1
Δtw : TWFMDT
Δtw < TWFMDT
Δtw = Δtw + δtw
2h
Δtw ≧ TWFMDT
Δtw : DTWFT
Δtw < DTWFT
Δtw = Δtw(last)
2h
Δtw ≧ DTWFT
Δt : TTFMDT
Δt < TTFMDT
Δt = Δt + δt
Convert Δt into units corresponding to input and write
2d
Δt ≧ TTFMDT
Δt : DTTFT
Δt < DTTFT
Δt = Δt (last)
Convert Δt into units corresponding to input and write
2d
Δt ≧ DTTFT
νo : ANLMDN
νo < ANLMDN
νo = νo + δνo
Convert Δt (first) into units corresponding to input; write νo and Δt (first)
2c
νo ≧ ANLMDN
νo : ANUOLT
νo < ANUOLT
νo = νo (last)
Convert Δt (first) into units corresponding to input; write νo and Δt (first)
2c
νo ≧ ANUOLT
Read in new input, beginning with input 1


11b
GUIDE = 2
Δt : TTFMDT
Δt < TTFMDT
Δt = Δt + δt
2h
Δt ≧ TTFMDT
Δt : DTTFT
Δt < DTTFT
Δt = Δt (last)
2h
Δt_T ≧ DTTFT
Δt_w : TWFMDT
Δt_w < TWFMDT
$\Delta t_w = \Delta t_w + \delta t_w$
Convert Δt_w into units corresponding to input and write
2g
Δt_w ≧ TWFMDT
Δt_w : DTWFT
Δt_w < DTWFT
$\Delta t_w = \Delta t_w$ (last)
Convert Δt_w into units corresponding to input and write
2g
Δt_w ≧ DTWFT
ν_o : ANLMDN
ν_o < ANLMDN
$\nu_o = \nu_o + \delta\nu_o$
Convert Δt_w (first) into units corresponding to input; write ν_o and Δt_w (first)
2f
ν_o ≧ ANLMDN
ν_o : ANUOLT
ν_o < ANUOLT
$\nu_o = \nu_o$ (last)
Convert Δt_w (first) into units corresponding to input; write ν_o and Δt_w (first)
2f
ν_o ≧ ANUOLT
Read in new input, beginning with input 1

APPENDIX E

FORTRAN PROGRAM

The FORTRAN program to determine the velocity increment required for rendezvous between two arbitrary elliptic orbits is as follows:

```
      WRITE(6,104)
      WRITE(6,102)
      J=7
      NJ=2
   10 READ(5,100)GUIDE,OPTION,DATAS
      READ(5,100)AMU,AT
      READ(5,100)AN,RMIN,ET,PHIO
      READ(5,100)DTWI,DTWF,DTTI,DTTF
      READ(5,100)DELTW,DELTT
      IF(NJ.GT.1)GO TO 48
      J=3
      GO TO 49
   48 J=J
   49 NJ=0
      IF(DATAS.GT.1.5)GO TO 11
      READ(5,100)EI,AI,OMEGAI,OMI
      READ(5,100)AII,ANU01,ANU0L,DELNU0
   24 TPER=6.2831853*SQRT((AT**2/AMU)*AT)
      VCT=SQRT(AMU/AT)
      WRITE(6,108)
      WRITE(6,106)ET,EI,AI,OMEGAI,OMI,AII
      WRITE(6,110)
      WRITE(6,106)PHIO,ANU01,ANU0L,AMU,AT,TPER,VCT
      WRITE(6,112)
      WRITE(6,126)DTTI,DTTF,DTWI,DTWF,RMIN,AN
      J=J+6
      IF(DATAS.GT.1.5)GO TO25
      WRITE(6,124)
      J=J+2
   55 WRITE(6,132)ANU01
C     IF DTTI IS INPUT AS ZERO, IT IS REASSIGNED A VALUE EQUIVALENT TO
C          THE INCREMENT SIZE.
      IF(DTTI.GT.0.0)GO TO 54
      ISIG=49
      CALL SPACE
      J=J+2
      DTTI=DELTT
      WRITE(6,162)DTTI
   54 CONTINUE
      IF(GUIDE.GT.1.5)GO TO 30
      IF(OPTION.LT.1.5)GO TO 50
      WRITE(6,140)DTTI
      GO TO 51
   50 WRITE(6,130)DTTI
   51 WRITE(6,116)
      GOTO31
   30 CONTINUE
      IF(OPTION.LT.1.5)GO TO 53
      WRITE(6,138)DTWI
      GO TO 52
   53 WRITE(6,128)DTWI
   52 WRITE(6,118)
   31 J=J+4
      IF(DATAS.GT.1.5)GO TO 29
```

```
      PI=AI*(1.0-EI**2)
      PT=1.0-ET**2
   26 PHI0=PHI0*.17453293E-01
      OMEGAI=OMEGAI*.17453293E-01
      OMI=OMI*.17453293E-01
      AII=AII*.17453293E-01
      ANU01=ANU01*.17453293E-01
      ANU0L=ANU0L*.17453293E-01
      DELNU0=DELNU0*.17453293E-01
      GO TO 200
   11 READ(5,100)X0,Y0,Z0
      READ(5,100)X0DT,Y0DT,Z0DT
      DELNU0=0.0
      PHI0=PHI0*.17453293E-01
      PT=1.0-ET**2
      CPHI0=COS(PHI0)
      SPHI0=SIN(PHI0)
      R0=PT/(1.0+ET*CPHI0)
      PHI0DT=SQRT(PT)/(R0**2)
      R0DT=(ET*SPHI0)/SQRT(PT)
      V01=R0DT+X0DT-PHI0DT*Y0
      V02=Y0DT+PHI0DT*(R0+X0)
      V03=Z0DT
      XI01=R0+X0
      XI02=Y0
      XI03=Z0
      XI0=SQRT(XI01**2+XI02**2+XI03**2)
      V0=SQRT(V01**2+V02**2+V03**2)
      AI=XI0/(2.0-XI0*V0**2)

C     IF AI IS LESS THAN OR EQUAL TO ZERO, A MESSAGE TO THIS EFFECT IS
C         WRITTEN AND TRANSFER IS TO THE BEGINNING OF THE PROGRAM,
C         WHERE A NEW SET OF DATA MAY BE READ IN.
      IF(AI)17,17,18
   17 TPER=6.2831853*SQRT((AT**2/AMU)*AT)
      VCT=SQRT(AMU/AT)
      WRITE(6,148)
      WRITE(6,106)PHI0,ET,AMU,AT,TPER,VCT
      WRITE(6,112)
      WRITE(6,126)DTTI,DTTF,DTWI,DTWF,RMIN,AH
      WRITE(6,114)
      WRITE(6,106)X0,Y0,Z0,X0DT,Y0DT,Z0DT
      WRITE(6,124)
      WRITE(6,144)
      WRITE(6,104)
      GO TO 10
   18 AH01=XI02*V03-XI03*V02
      AH02=XI03*V01-XI01*V03
      AH03=XI01*V02-XI02*V01
      PI=AH01**2+AH02**2+AH03**2
      AH0=SQRT(PI)
      XI0DT=(V01*XI01+V02*XI02+V03*XI03)/XI0
      PIDAI=PI/AI
```

```
C      TEST TO INSURE THAT EI IS NOT EQUAL TO THE SQRT OF A NEG QUANTITY
C           DUE TO SLIGHT COMPUTING INACCURACY ON PI/AI.  ASSIGN VALUES
C           FOR COS(NUO) AND SIN(NUO) IF THE QUANTITY UNDER THE RADICAL
C           IS NEGATIVE.
       IF(PIDAI.LT.1.0)GO TO 19
       EI=0.0
       CNU0=1.0
       SNU0=0.0
       GO TO 28
    19 CONTINUE
       EI=SQRT(1.0-(PI/AI))
       CNU0=((PI/XI0)-1.0)/EI
       SNU0=(SQRT(PI))*XI0DT/EI
    28 CONTINUE
       ANU0=ATAN2(SNU0,CNU0)
       IF(SNU0.LT.0.0.AND.CNU0.GT.0.0)ANU0=6.2831853+ANU0
       IF(SNU0.LT.0.0.AND.CNU0.LT.0.0)ANU0=6.2831853+ANU0
    13 CONTINUE
       ANU01=ANU0
       ANU0L=ANU0
    15 CII=AH03/AH0
       IF(CII.GE.0.99999995)GO TO 16
       AII=ATAN(SQRT(1.0-CII**2)/CII)
       IF(CII.LT.0.0)AII=3.1415927+AII
       SAII=SIN(AII)
       COMEGI=-(AH02*CPHI0+AH01*SPHI0)/(AH0*SAII)
       SOMEGI=(AH01*CPHI0-AH02*SPHI0)/(AH0*SAII)
       OMEGAI=ATAN2(SOMEGI,COMEGI)
       IF(SOMEGI.LT.0.0.AND.COMEGI.GT.0.0)OMEGAI=6.2831853+OMEGAI
       IF(SOMEGI.LT.0.0.AND.COMEGI.LT.0.0)OMEGAI=6.2831853+OMEGAI
    20 CONTINUE
       COPN=(XI02*AH01-XI01*AH02)/(XI0*AH0*SAII)
       SOPN=XI03/(XI0*SAII)
       GO TO 21
    16 AII=0.0
       OMEGAI=0.0
       COPN=(XI01*CPHI0-XI02*SPHI0)/XI0
       SOPN=(XI01*SPHI0+XI02*CPHI0)/XI0
    21 OPN=ATAN2(SOPN,COPN)
       IF(SOPN.LT.0.0.AND.COPN.GT.0.0)OPN=6.2831853+OPN
       IF(SOPN.LT.0.0.AND.COPN.LT.0.0)OPN=6.2831853+OPN
    23 CONTINUE
       OMI=OPN-ANU0
C      CONVERT ANGLES IN RADIANS TO DEGREES FOR PRINTING
       PHI0=PHI0/.17453293E-01
       OMEGAI=OMEGAI/.17453293E-01
       OMI=OMI/.17453293E-01
       AII=AII/.17453293E-01
       ANU01=ANU01/.17453293E-01
       ANU0L=ANU0L/.17453293E-01
       DELNU0=DELNU0/.17453293E-01
       GO TO 24
    25 WRITE(6,114)
       WRITE(6,106)X0,Y0,Z0,X0DT,Y0DT,Z0DT
```

```
      WRITE(6,124)
      J=J+4
      GO TO 55
C     CHANGE TIMES TO DIMENSIONLESS QUANTITIES
  200 IF(OPTION.LT.1.5)GO TO 201
      IF(OPTION.LT.2.5)GO TO 202
      GO TO 203
  202 CMIN=60./TPER
      DTWI=DTWI*CMIN
      DTWF=DTWF*CMIN
      DELTW=DELTW*CMIN
      DTTI=DTTI*CMIN
      DTTF=DTTF*CMIN
      DELTT=DELTT*CMIN
      GO TO 201
  203 CDAYS=86400./TPER
      DTWI=DTWI*CDAYS
      DTWF=DTWF*CDAYS
      DELTW=DELTW*CDAYS
      DTTI=DTTI*CDAYS
      DTTF=DTTF*CDAYS
      DELTT=DELTT*CDAYS
  201 CONTINUE
      IF(DTWF.GT.0.0)GO TO 204
      TESTTW=.0000001
      GO TO 205
  204 TESTTW=.5E-06*DTWF
  205 IF(DTTF.GT.0.0)GO TO 206
      TESTTT=.0000001
      GO TO 207
  206 TESTTT=.5E-06*DTTF
  207 CONTINUE
      TWFMDT=DTWF-1.5*DELTW-TESTTW
      TTFMDT=DTTF-1.5*DELTT-TESTTT
      ANLMDN=ANUOL-1.5*DELNU0-.0000001
      DTWFT=DTWF-TESTTW
      DTTFT=DTTF-TESTTT
      ANUOLT=ANUOL-.0000001
      IF(GUIDE-1.5)39,43,43
   39 ANU0=ANU01
   40 DELTTT=DTTI
   41 DELTTW=DTWI
      GO TO42
   43 ANU0=ANU01
   44 DELTTW=DTWI
   45 DELTTT=DTTI
C     CRIT IS A VALUE USED TO TEST THE ITERATION SCHEMES FOR CONVERGENCE
   42 CRIT=.5E-06*DELTTT
      CRIRHO=1.0E-06*AI
      COMMONJ,GUIDE,ISIG
      GAMMA=1.0
   61 ET0=AAA(ET,PHI0,GAMMA)
      AMT0=ET0-ET*(SIN(ET0))
      AMTF=AMT0+6.2831853*(DELTTT+DELTTW)
```

```
      CALL ANOM(ET,AMTF,ETF)
   63 PHIF=AAA(-ET,ETF,GAMMA)
      CPHIF=COS(PHIF)
      SPHIF=SIN(PHIF)
      RF=PT/(1.0+ET*CPHIF)
      PHIDTF=(SQRT(PT))/(RF**2)
      RDTF=ET*SPHIF/SQRT(PT)
   65 EI0=AAA(EI,ANU0,GAMMA)
      AMI0=EI0-EI*(SIN(EI0))
      AMII=AMI0+((6.2831853*DELTTW)/(AI**1.5))
      CALL ANOM(EI,AMII,EII)
   67 ANUI=AAA(-EI,EII,GAMMA)
      XII=PI/(1.0+EI*COS(ANUI))
      ANUDTI=SQRT(PI)/(XII**2)
      XIDTI=EI*SIN(ANUI)/(SQRT(PI))
C     DETERMINATION OF DELTTH AND I OF TRANSFER ORBIT
  300 OIPNI=ANUI+OMI
      COIPNI=COS(OIPNI)
      SOIPNI=SIN(OIPNI)
      COMEGI=COS(OMEGAI)
      SOMEGI=SIN(OMEGAI)
      CII=COS(AII)
      SII=SIN(AII)
      XII1=XII*(COMEGI*COIPNI-CII*SOMEGI*SOIPNI)
      XII2=XII*(SOMEGI*COIPNI+CII*COMEGI*SOIPNI)
      XII3=XII*(SII*SOIPNI)
      SUB1=XII3/(XII1*SPHIF-XII2*CPHIF)
      BI=ATAN(SUB1)
      ASOLI=ABS(BI)
      IF(ASOLI-1.5690510)306,305,305
  305 CONTINUE
      ISIG=48
      CALL SPACE
      WRITE(6,150)
      J=J+2
      GO TO 1028
  306 CONTINUE
      CDTH=(XII1*CPHIF+XII2*SPHIF)/XII
      SDTH=(XII1*SPHIF-XII2*CPHIF)/(XII*COS(BI))
      DELTTH=ATAN2(SDTH,CDTH)
      IF(SDTH.LT.0.0.AND.CDTH.GT.0.0)DELTTH=6.2831853+DELTTH
      IF(SDTH.LT.0.0.AND.CDTH.LT.0.0)DELTTH=6.2831853+DELTTH
  302 CONTINUE
C     COMPUTATION OF PARABOLIC TIME
      RHO2=AMAX1(RF,XII)
      RHO1=AMIN1(RF,XII)
      IF(DELTTH.GT.0.1745E-04.AND.DELTTH.LT.6.2831679)GO TO 308
      DEGDTH=DELTTH/.17453293E-01
      ISIG=47
      CALL SPACE
      J=J+1
      WRITE(6,160)DEGDTH
      A=1.0
      P=PT
```

APPENDIX E

```
      TH1=PHIF-DELTTH
      TH2=PHIF
      E=ET
      GO TO 800
  308 CONTINUE
      C=SQRT(RHO1**2+RHO2**2-2.0*RHO1*RHO2*CDTH)
      SPSI=(RHO1-RHO2*CDTH)/C
      CPSI=(RHO2*SDTH)/C
      PSI=ATAN2(SPSI,CPSI)
      IF(SPSI.LT.0.0.AND.CPSI.LT.0.0)PSI=6.2831853+PSI
  304 CONTINUE
      SALP=(RHO2-RHO1)/C
      IF(SALP.GE.1.0)GO TO 311
      ALPHA=ATAN(SALP/SQRT(1.0-SALP**2))
      GO TO 312
  311 ALPHA=1.5707963
  312 CONTINUE
      THP=ALPHA-PSI
      PP=RHO1*(1.0+COS(THP))
      TTHP=(SIN(THP/2.))/(COS(THP/2.))
      TTHPDT=(SIN((THP+DELTTH)/2.))/(COS((THP+DELTTH)/2.))
      TP1=((PP**1.5)*((TTHP**3)/6.+TTHP/2.))/6.2831853
      TP2=((PP**1.5)*((TTHPDT**3)/6.+TTHPDT/2.))/6.2831853
      DELTTP=TP2-TP1
      THPMAX=3.1415927-(ALPHA+PSI)
      SUB12=DELTTP-DELTTT
      ASUB12=ABS(SUB12)
      IF(ASUB12-CRIT)37,37,38
C     IF DELTTP EQUALS DELTTT,THE ORBIT IS PARABOLIC
   37 E=1.0
      P=PP
      TH1=THP
      TH2=TH1+DELTTH
C     A IS INFINITE, IS SET EQUAL TO 99.999 FOR PRINTING PURPOSES
      A=99.999
      GO TO 800
C     IF DELTTP IS GREATER THAN DELTTT,THE ORBIT IS HYPERBOLIC
C     IF DELTTP IS LESS THAN DELTTT,THE ORBIT IS ELLIPTIC
   38 IF(DELTTP-DELTTT)400,37,600
  400 SUB13=RHO1-RHO2
      ASUB13=ABS(SUB13)
      IF(ASUB13-CRIRHO)500,500,401
C     ELLIPTIC LOOP   RHO2 NOT EQUAL TO RHO1
  401 DELTH=(3.1415927-2.*(THP+PSI))/AN
      TH1=THP+DELTH
  402 TH2=TH1+DELTTH
     *E=(RHO2-RHO1)/(RHO1*COS(TH1)-RHO2*COS(TH2))
      DIFF=THPMAX-TH1
      IF(DIFF.LE.0.0)GO TO 405
      IF(E.GE.1.0)GO TO 405
      P=RHO1*(1.0+E*COS(TH1))
      A=P/(1.0-E**2)
      GAMMA=1.0
      E1=AAA(E,TH1,GAMMA)
```


See errata sheet

```
      E2=AAA(E,TH2,GAMMA)
      DT=((A**1.5)*(E2-E1-E*(SIN(E2)-SIN(E1))))/6.2831853
      SUB14=DT-DELTTT
      ASUB14=ABS(SUB14)
      IF(ASUB14-CRIT)800,800,403
  403 IF(SUB14)404,800,405
  404 TH1=TH1+DELTH
      GO TO 402
  405 DELTH=DELTH/2.
      IF(DELTH-.0000001)800,800,408
  408 TH1=TH1-DELTH
      GO TO 402
C     ELLIPTIC LOOP   RHO2 EQUAL TO RHO1
  500 DELTTC=(DELTTH*(RHO1**1.5))/6.2831853
      DELE=1.0/AH
      E=DELE
      SUB15=DELTTC-DELTTT
      IF(SUB15)502,501,501
  501 P=RHO1*(1.0+E*(COS(PSI)))
      A=P/(1.0-E**2)
      TH1=-1.0*PSI
      TH2=TH1+DELTTH
      GAMMA=1.0
      E1=AAA(E,TH1,GAMMA)
      DT=-((A**1.5)*(E1-E*SIN(E1)))/3.1415927
      SUB16=DT-DELTTT
      ASUB16=ABS(SUB16)
      IF(ASUB16-CRIT)800,800,503
  503 IF(SUB16)504,800,505
  504 DELE=DELE/2.
      IF(DELE-.0000001)800,800,512
  512 E=E-DELE
      GO TO 501
  505 E=E+DELE
      GO TO 501
  502 P=RHO1*(1.0-E*(COS(PSI)))
      A=P/(1.0-E**2)
      TH1=3.1415927-PSI
      TH2=TH1+DELTTH
      GAMMA=1.0
      E1=AAA(E,TH1,GAMMA)
      AM1=E1-E*(SIN(E1))
      DT=((A**1.5)*(3.1415927-AM1))/3.1415927
      SUB17=DT-DELTTT
      ASUB17=ABS(SUB17)
      IF(ASUB17-CRIT)800,800,506
  506 IF(SUB17)507,800,508
  507 E=E+DELE
      GO TO 502
  508 DELE=DELE/2.
      IF(DELE-.0000001)800,800,514
  514 E=E-DELE
      GO TO 502
```

APPENDIX E

```
  600 SUB18=RHO1-RHO2
      ASUB18=ABS(SUB18)
      IF(ASUB18-CRIRHO)700,700,601
C     HYPERBOLIC LOOP   RHO2 NOT EQUAL TO RHO1
  601 DELTH=(THP+PSI)/AH
  610 TH1=THP-DELTH
  602*TH2=TH1+DELTTH
      E=(RHO2-RHO1)/(RHO1*(COS(TH1))-RHO2*(COS(TH2)))
      CTH1=COS(TH1)
      OPECT=1.0+E*CTH1
      IF(OPECT.GT.0.0)GO TO 608
  609 DELTH=DELTH/2.
      IF(DELTH.GT.0.1E-08)GO TO 612
C     IF THE CASE IS A LIMITING HYPERBOLIC ORBIT WITH E APPROACHING
C          (-1.0)/COS(DELTTH/2.0) WRITE A MESSAGE AND DISCONTINUE
C          COMPUTATION FOR THAT CASE.
      ISIG=49
      CALL SPACE
      J=J+2
      E=(-1.0)/COS(DELTTH/2.0)
      IF(GUIDE.GT.1.5)GO TO 611
      CALL CONVRT(DELTTW,CDTW,OPTION,TPER)
      WRITE(6,164)CDTW,E
      GO TO 1028
  611 CALL CONVRT(DELTTT,CDTT,OPTION,TPER)
      WRITE(6,166)CDTT,E
      GO TO 1028
  612 CONTINUE
      TH1=TH1+DELTH
      GO TO 602
  608 IF(E.LE.1.0)GO TO 604
      P=RHO1*(1.0+E*(COS(TH1)))
      A=P/(1.0-E**2)
      GAMMA=-1.0
      F1=AAA(E,TH1,GAMMA)
      F2=AAA(E,TH2,GAMMA)
      TANF1=(SIN(F1))/(COS(F1))
      SUB19=F1/2.+3.1415927/4.
      TS19=(SIN(SUB19))/(COS(SUB19))
      T1=(((-A)**1.5)*(E*TANF1-ALOG(TS19)))/6.2831853
      TANF2=(SIN(F2))/(COS(F2))
      SUB20=F2/2.+3.1415927/4.
      TS20=(SIN(SUB20))/(COS(SUB20))
      T2=(((-A)**1.5)*(E*TANF2-ALOG(TS20)))/6.2831853
      DT=T2-T1
      SUB21=DT-DELTTT
      ASUB21=ABS(SUB21)
      IF(ASUB21-CRIT)800,800,603
```

* See errata sheet

```
  603 IF(SUB21)604,800,605
  604 DELTH=DELTH/2.
      IF(DELTH-.0000001)800,800,607
  607 CONTINUE
      TH1=TH1+DELTH
      GO TO 602
  605 TH1=TH1-DELTH
      GO TO 602
C     HYPERBOLIC LOOP    XM1 IS EQUAL TO XM 2
  700 DELE=1./AN
      E=1.+DELE
      TH1=-PSI
      CTH1=COS(TH1)
      TH2=TH1+DELTTH
  701 P=RHO1*(1.+E*CTH1)
      A=P/(1.-E**2)
      GAMMA=-1.0
      F1=AAA(E,TH1,GAMMA)
      TF1=(SIN(F1))/(COS(F1))
      SUB22=F1/2.+3.1415927/4.
      TS22=(SIN(SUB22))/(COS(SUB22))
      DT=-(((-A)**1.5)*(E*TF1-ALOG(TS22)))/3.1415927
      SUB23=DT-DELTTT
      ASUB23=ABS(SUB23)
      IF(ASUB23-CRIT)800,800,702
  702 IF(SUB23)703,800,704
  703 DELE=DELE/2.
      IF(DELE-.0000001)800,800,707
  707 E=E-DELE
      GO TO 701
  704 E=E+DELE
      GO TO 701
  800 IF(TH1)801,802,802
  801 RHOMIN=P/(1.+E)
      GO TO 803
  802 RHOMIN=RHO1
      GO TO 803
  803 K=0
      IF(RHOMIN-RMIN)804,805,805
  804 K=2
  805 IF(RF-XII)806,807,807
  806 RHOI=RHO2
      RHOF=RHO1
      THI=6.2831853-TH2
      THF=THI+DELTTH
      GO TO 808
  807 RHOI=RHO1
      RHOF=RHO2
      THI=TH1
      THF=TH2
  808 SQP=SQRT(P)
      THDTI=SQP/(RHOI**2)
      RHODTI=E*SIN(THI)/SQP
      THDTF=SQP/(RHOF**2)
```

```
      RHODTF=E*SIN(THF)/SQP
    * CALL SIX(A11,A12,A21,A22,A31,A32,OM,OMEGAI,AII)
  809 ABI=ABS(BI)
      IF(ABI-5.E-08)810,810,811
  810 OMEGA=0.0
      OM=PHIF-THF
      GO TO 900
  811 IF(BI)813,810,812
  812 OMEGA=PHIF+3.1415927
      OM=3.1415927-THF
      GO TO 900
  813 OMEGA=PHIF
      OM=6.2831853-THF
  900 CALL SIX(B11,B12,B21,B22,B31,B32,OM,OMEGA,BI)
      XIIDT1=XIDTI*(COS(ANUI))-XII*ANUDTI*(SIN(ANUI))
      XIIDT2=XIDTI*(SIN(ANUI))+XII*ANUDTI*(COS(ANUI))
      RODTI1=RHODTI*(COS(THI))-RHOI*THDTI*(SIN(THI))
      RODTI2=RHODTI*(SIN(THI))+RHOI*THDTI*(COS(THI))
      RODTF1=RHODTF*(COS(THF))-RHOF*THDTF*(SIN(THF))
      RODTF2=RHODTF*(SIN(THF))+RHOF*THDTF*(COS(THF))
      RDTF1=RDTF*(COS(PHIF))-RF*PHIDTF*(SIN(PHIF))
      RDTF2=RDTF*(SIN(PHIF))+RF*PHIDTF*(COS(PHIF))
  901 CALL VELOC(V11,V12,V13,A11,A12,A21,A22,A31,A32,XIIDT1,XIIDT2)
      CALL VELOC(V21,V22,V23,B11,B12,B21,B22,B31,B32,RODTI1,RODTI2)
      CALL VELOC(V31,V32,V33,B11,B12,B21,B22,B31,B32,RODTF1,RODTF2)
      V41=RDTF1
      V42=RDTF2
      V43=0.0
 1001 DV11=V21-V11
      DV12=V22-V12
      DV13=V23-V13
      DV21=V41-V31
      DV22=V42-V32
      DV23=V43-V33
 1002 DELV1=SQRT(DV11**2+DV12**2+DV13**2)
      DELV2=SQRT(DV21**2+DV22**2+DV23**2)
      DELV=DELV1+DELV2
      PHIF=PHIF/.17453293E-01
      ANUI=ANUI/.17453293E-01
      THI=THI/.17453293E-01
      DELTTH=DELTTH/.17453293E-01
      BI=BI/.17453293E-01
      CALL CONVRT(DELTTT,DELTTT,OPTION,TPER)
      CALL CONVRT(DELTTW,DELTTW,OPTION,TPER)
      ISIG=49
      CALL SPACE
      J=J+2
      IF(GUIDE.GT.1.5)GO TO 1030
      IF(K.GT.1)GO TO 1032
      IF(OPTION.GT.1.5)GO TO 1040
      WRITE(6,120)DELTTW,PHIF,ANUI,E,A,BI,THI,DELTTH,RHOMIN,DV11,DV12,DV
     113,DV21,DV22,DV23,DELV1,DELV2,DELV
      GO TO 1031
```

* [illegible]

```
1040 WRITE(6,134)DELTTW,PHIF,ANUI,E,A,BI,THI,DELTTH,RHOMIN,DV11,DV12,DV
    113,DV21,DV22,DV23,DELV1,DELV2,DELV
     GO TO 1031
1032 IF(OPTION.GT.1.5)GO TO 1041
     WRITE(6,122)DELTTW,PHIF,ANUI,E,A,BI,THI,DELTTH,RHOMIN,DV11,DV12,DV
    113,DV21,DV22,DV23,DELV1,DELV2,DELV
     GO TO 1031
1041 WRITE(6,136)DELTTW,PHIF,ANUI,E,A,BI,THI,DELTTH,RHOMIN,DV11,DV12,DV
    113,DV21,DV22,DV23,DELV1,DELV2,DELV
     GO TO 1031
1030 IF(K.GT.1)GO TO 1033
     IF(OPTION.GT.1.5)GO TO 1042
     WRITE(6,120)DELTTT,PHIF,ANUI,E,A,BI,THI,DELTTH,RHOMIN,DV11,DV12,DV
    113,DV21,DV22,DV23,DELV1,DELV2,DELV
     GO TO 1031
1042 WRITE(6,134)DELTTT,PHIF,ANUI,E,A,BI,THI,DELTTH,RHOMIN,DV11,DV12,DV
    113,DV21,DV22,DV23,DELV1,DELV2,DELV
     GO TO 1031
1033 IF(OPTION.GT.1.5)GO TO 1043
     WRITE(6,122)DELTTT,PHIF,ANUI,E,A,BI,THI,DELTTH,RHOMIN,DV11,DV12,DV
    113,DV21,DV22,DV23,DELV1,DELV2,DELV
     GO TO 1031
1043 WRITE(6,136)DELTTT,PHIF,ANUI,E,A,BI,THI,DELTTH,RHOMIN,DV11,DV12,DV
    113,DV21,DV22,DV23,DELV1,DELV2,DELV
1031 CONTINUE
C    CONVERT BACK TO DIMENSIONLESS TIME
     IF(OPTION.LT.1.5)GO TO 1028
     IF(OPTION.LT.2.5)GO TO 1072
     DELTTT=86400.*DELTTT/TPER
     DELTTW=86400.*DELTTW/TPER
     GO TO 1028
1072 DELTTT=60.*DELTTT/TPER
     DELTTW=60.*DELTTW/TPER
1028 CONTINUE
     IF(GUIDE-1.5)1003,1015,1015
1003 IF(DELTTW-TWFNDT)1004,1005,1005
1004 DELTTW=DELTTW+DELTW
     NO=0
     GO TO42
1005 IF(NO.GT.1)GO TO 1007
     IF(DELTTW-DTWFT)1006,1007,1007
1006 DELTTW=DTWF
     NO=2
     GO TO 42
1007 IF(DELTTT-TTFNDT)1008,1009,1009
1008 DELTTT=DELTTT+DELTT
     NO=0
     NO1=0
     ISIG=48
     CALL SPACE
     WRITE(6,124)
     CALL CONVRT(DELTTT,CDTT,OPTION,TPER)
     IF(OPTION.GT.1.5)GO TO 1060
     WRITE(6,130)CDTT
```

```
      GO TO 1061
 1060 WRITE(6,140)CDTT
 1061 J=J+3
      GO TO 41
 1009 IF(NO1.GT.1)GO TO 1011
      IF(DELTTT-DTTFT)1010,1011,1011
 1010 DELTTT=DTTF
      NO=0
      NO1=2
      ISIG=48
      CALL SPACE
      WRITE(6,124)
      CALL CONVRT(DELTTT,CDTT,OPTION,TPER)
      IF(OPTION.GT.1.5)GO TO 1062
      WRITE(6,130)CDTT
      GO TO 1063
 1062 WRITE(6,140)CDTT
 1063 J=J+3
      GO TO 41
 1011 IF(ANU0-ANLMDN)1012,1013,1013
 1012 ANU0=ANU0+DELNU0
      NO=0
      NO1=0
      NO2=0
      ISIG=47
      CALL SPACE
      WRITE(6,124)
      ANU0=ANU0/.17453293E-01
      WRITE(6,132)ANU0
      CALL CONVRT(DTTI,CDTTI,OPTION,TPER)
      IF(OPTION.GT.1.5)GO TO 1064
      WRITE(6,130)CDTTI
      GO TO 1065
 1064 WRITE(6,140)CDTTI
 1065 ANU0=ANU0*.17453293E-01
      J=J+3
      GO TO 40
 1013 IF(NO2.GT.1)GO TO 12
      IF(ANU0-ANU0LT)1014,12,12
 1014 ANU0=ANU0L
      NO=0
      NO1=0
      NO2=2
      ISIG=47
      CALL SPACE
      WRITE(6,124)
      ANU0=ANU0/.17453293E-01
      WRITE(6,132)ANU0
      CALL CONVRT(DTTI,CDTTI,OPTION,TPER)
      IF(OPTION.GT.1.5)GO TO 1066
      WRITE(6,130)CDTTI
      GO TO 1067
 1066 WRITE(6,140)CDTTI
 1067 ANU0=ANU0*.17453293E-01
```

```
      J=J+3
      GO TO 40
 1015 IF(DELTTT-TTFMDT)1016,1017,1017
 1016 DELTTT=DELTTT+DELTT
      NO=0
      GO TO 42
 1017 IF(NO.GT.1)GO TO 1019
      IF(DELTTT-DTTFT)1018,1019,1019
 1018 DELTTT=DTTF
      NO=2
      GO TO 42
 1019 IF(DELTTW-TWFMDT)1020,1021,1021
 1020 DELTTW=DELTTW+DELTW
      NO=0
      NO1=0
      ISIG=48
      CALL SPACE
      WRITE(6,124)
      CALL CONVRT(DELTTW,CDTW,OPTION,TPER)
      IF(OPTION.GT.1.5)GO TO 1080
      WRITE(6,128)CDTW
      GO TO 1081
 1080 WRITE(6,138)CDTW
 1081 J=J+2
      GO TO 45
 1021 IF(NO1.GT.1)GO TO 1023
      IF(DELTTW-DTWFT)1022,1023,1023
 1022 DELTTW=DTWF
      NO=0
      NO1=2
      ISIG=48
      CALL SPACE
      WRITE(6,124)
      CALL CONVRT(DELTTW,CDTW,OPTION,TPER)
      IF(OPTION.GT.1.5)GO TO 1082
      WRITE(6,128)CDTW
      GO TO 1083
 1082 WRITE(6,138)CDTW
 1083 J=J+2
      GO TO 45
 1023 IF(ANU0-ANLMDN)1024,1025,1025
 1024 ANU0=ANU0+DELNU0
      NO=0
      NO1=0
      NO2=0
      ISIG=47
      CALL SPACE
      WRITE(6,124)
      ANU0=ANU0/.17453293E-01
      WRITE(6,132)ANU0
      CALL CONVRT(DTWI,CDTWI,OPTION,TPER)
      IF(OPTION.GT.1.5)GO TO 1084
      WRITE(6,128)CDTWI
```

APPENDIX E

```
      GO TO 1085
 1084 WRITE(6,138)CDTWI
 1085 ANU0=ANU0*.17453293E-01
      J=J+3
      GO TO 44
 1025 IF(NO2.GT.1)GO TO 12
      IF(ANU0-ANU0LT)1026,12,12
 1026 ANU0=ANU0L
      NO=0
      NO1=0
      NO2=2
      ISIG=47
      CALL SPACE
      WRITE(6,124)
      ANU0=ANU0/.17453293E-01
      WRITE(6,132)ANU0
      CALL CONVRT(DTWI,CDTWI,OPTION,TPER)
      IF(OPTION.GT.1.5)GO TO 1086
      WRITE(6,128)CDTWI
      GO TO 1087
 1086 WRITE(6,138)CDTWI
 1087 ANU0=ANU0*.17453293E-01
      J=J+3
      GO TO 44
   12 WRITE(6,104)
      GO TO 10
  100 FORMAT(4E18.8)
  102 FORMAT(//4X42HVELOCITY INCREMENT REQUIRED FOR RENDEZVOUS1X37HBETWE
     1EN TWO ARBITRARY ELLIPTIC ORBITS//)
  104 FORMAT(1H1)
  106 FORMAT(7E18.8)
  108 FORMAT(12X2HET16X2HEI16X2HAI12X6HOMEGAI15X3HOMI16X2HII)
  110 FORMAT(10X4HPHI014X4HNU0114X4HNU0L16X2HMU16X2HAT14X4HTPER15X3HVCT)
  112 FORMAT(10X4HDTTI14X4HDTTF14X4HDTWI14X4HDTWF32X4HRMIN17X1HH)
  114 FORMAT(12X2HX016X2HY016X2HZ014X4HX0DT14X4HY0DT14X4HZ0DT)
  116 FORMAT(1X6HDELTTW3X4HPHIF4X3HNUI6X1HE6X1HA6X1HI5X3HTHI1X6HDELTTH1X
     14HRHOM4X4HDV1X4X4HDV1Y4X4HDV1Z4X4HDV2X4X4HDV2Y4X4HDV2Z2X5HDELV12X
     25HDELV23X4HDELV//)
  118 FORMAT(1X6HDELTTT3X4HPHIF4X3HNUI6X1HE6X1HA6X1HI5X3HTHI1X6HDELTTH1X
     14HRHOM4X4HDV1X4X4HDV1Y4X4HDV1Z4X4HDV2X4X4HDV2Y4X4HDV2Z2X5HDELV12X
     25HDELV23X4HDELV//)
  120 FORMAT(F7.4,2F7.2,F7.4,F7.3,F7.2,F8.2,F7.2,F5.2,6F8.5,3F7.4//)
  122 FORMAT(F7.4,2F7.2,F7.4,F7.3,F7.2,F8.2,F7.2,F5.2,6F8.5,3F7.4,1H*//)
  124 FORMAT(1X//)
  126 FORMAT(4E18.8,E36.8,E18.8)
  128 FORMAT(1X8HDELTTW= F7.4)
  130 FORMAT(1X8HDELTTT= F7.4)
  132 FORMAT(4X5HNU0= F7.2)
  134 FORMAT(F7.1,2F7.2,F7.4,F7.3,F7.2,F8.2,F7.2,F5.2,6F8.5,3F7.4//)
  136 FORMAT(F7.1,2F7.2,F7.4,F7.3,F7.2,F8.2,F7.2,F5.2,6F8.5,3F7.4,1H*//)
  138 FORMAT(1X8HDELTTW= F7.1)
  140 FORMAT(1X8HDELTTT= F7.1)
  144 FORMAT(6X20HAI IS LESS THAN ZERO//)
  148 FORMAT(10X4HPHI016X2HET16X2HMU16X2HAT14X4HTPER15X3HVCT)
```

APPENDIX E

```
150 FORMAT(4X41HI EQUAL +/-90 DEGREES NOT ACCEPTABLE DATA//)
160 FORMAT(10X19HDELTA THETA EQUALS F6.1,1X39HDEGREES.  DO NOT CONSIDE
   1R UNLESS ORBITS1X25HINTERSECT, IN WHICH CASE,)
162 FORMAT(6X48HDTTI INPUT EITHER AS ZERO OR A NEGATIVE QUANTITY1X23HB
   1UT EQUATED TO DELTT ( F10.4,1X33H)  BY PROGRAM.  RECONSIDER INPUT.
   2//)
164 FORMAT(10X20HTHIS CASE (DELTTW = F10.4,1X37H) IS A LIMITING HYPERB
   1OLIC ORBIT WITH1X14HE APPROACHING F10.5//)
166 FORMAT(10X20HTHIS CASE (DELTTT = F10.4,1X37H) IS A LIMITING HYPERB
   1OLIC ORBIT WITH1X14HE APPROACHING F10.5//)
    END
```

```
      FUNCTION AAA(ALPHA,BETA,GAMMA)
C     GAMMA EQUAL 1,EXCEPT FOR HYPERBOLIC ORBITS WHEN IT IS -1
      PINUM=3.1415927
      BETA2=BETA/2.0
      ABETA2=ABS(BETA2)
      SBETA2=SIN(BETA2)
      IF(SBETA2.GT.0.0.AND.ABETA2.GT.1.5707789.AND.ABETA2.LT.1.5708137)G
     1O TO 1
      IF(SBETA2.LT.0.0.AND.ABETA2.GT.1.5707789.AND.ABETA2.LT.1.5708137)G
     1O TO 2
      SUB1=(SQRT(GAMMA*(1.0-ALPHA)/(1.0+ALPHA)))*(SBETA2/(COS(BETA2)))
      APR=2.*ATAN(SUB1)
      GO TO 3
    1 APR=3.1415927
      GO TO 3
    2 APR=-3.1415927
    3 N=(BETA+PINUM)/(2.*PINUM)
      AN=N
      AAA=APR+2.*AN*PINUM
      RETURN
      END
```

```
      SUBROUTINE VELOC(Y11,Y12,Y13,H11,H12,H21,H22,H31,H32,S1,S2)
      Y11=H11*S1+H12*S2
      Y12=H21*S1+H22*S2
      Y13=H31*S1+H32*S2
      RETURN
      END
```

APPENDIX E

```
      SUBROUTINE SIX(W11,W12,W21,W22,W31,W32,U1,U2,U3)
      CU1=COS(U1)
      SU1=SIN(U1)
      CU2=COS(U2)
      SU2=SIN(U2)
      CU3=COS(U3)
      SU3=SIN(U3)
      W11=CU1*CU2-CU3*SU2*SU1
      W12=-SU1*CU2-CU3*SU2*CU1
      W21=CU1*SU2+CU3*CU2*SU1
      W22=-SU1*SU2+CU3*CU2*CU1
      W31=SU3*SU1
      W32=SU3*CU1
      RETURN
      END
```

```
      SUBROUTINE ANOM(ECCEN,AMANOM,EANOM)
      BARM=AMANOM
      E=ECCEN
      EA0=BARM+E*SIN(BARM)+.5*E**2*SIN(2.*BARM)
      EA=EA0
   16 AMA=EA-E*SIN(EA)
      DELMA=BARM-AMA
      DELEA=DELMA/(1.-E*COS(EA))
      EA1=EA+DELEA
      ADELEA=ABS(DELEA)
      CONTST=(EA1-EA)/EA1
      ACT=ABS(CONTST)
      IF(ACT-.1E-06)17,15,15
   15 EA=EA1
      GO TO 16
   17 EANOM=EA
      RETURN
      END
```

```
      SUBROUTINE SPACE
      COMMON J,GUIDE,ISIG
      IF(ISIG-J)10,10,11
   10 WRITE(6,101)
```

```
      IF(GUIDE.GT.1.5)GO TO 12
      WRITE(6,103)
      GO TO 13
   12 WRITE(6,104)
   13 J=5
      RETURN
   11 CONTINUE
      RETURN
  101 FORMAT(1H1)
  103 FORMAT(1X6HDELTTW3X4HPHIF4X3HNUI6X1HE6X1HA6X1HI5X3HTHI1X6HDELTTH1X
     14HRHOM4X4HDV1X4X4HDV1Y4X4HDV1Z4X4HDV2X4X4HDV2Y4X4HDV2Z2X5HDELV12X
     25HDELV23X4HDELV//)
  104 FORMAT(1X6HDELTTT3X4HPHIF4X3HNUI6X1HE6X1HA6X1HI5X3HTHI1X6HDELTTH1X
     14HRHOM4X4HDV1X4X4HDV1Y4X4HDV1Z4X4HDV2X4X4HDV2Y4X4HDV2Z2X5HDELV12X
     25HDELV23X4HDELV//)
      END
```

```
      SUBROUTINE CONVRT(TIME,CTIME,OPTION,TPER)
C     TO CONVERT TIME FROM DIMENSIONLESS QUANTITIES TO UNITS
C            CORRESPONDING TO INPUT
      IF(OPTION.LT.1.5)GO TO 1
      IF(OPTION.LT.2.5)GO TO 2
      CTIME=TIME*TPER/86400.
      RETURN
    1 CTIME=TIME
      RETURN
    2 CTIME=TIME*TPER/60.
      RETURN
      END
```

REFERENCES

1. Battin, Richard H.: The Determination of Round-Trip Planetary Reconnaissance Trajectories. J. Aero Space Sci., vol. 26, no. 9, Sept. 1959, pp. 545-567.
2. Breakwell, John V.; Gillespie, Rollin W.; and Ross, Stanley: Researches in Interplanetary Transfer. [Preprint] 954-59, Am. Rocket Soc., Nov. 1959.
3. Lascody, D. N.: Analytical Determination of Three Dimensional Interplanetary Transfers. Proceedings XIIIth International Astronautical Congress (Varna), vol. II, N. Boneff and I. Hersey, eds., Springer-Verlag, 1964, pp. 571-594.
4. Anon.: Planetary Flight Handbook. Part 1 – Speed Contours and Auxiliary Graphs for Manned Missions to Mars and Venus. NASA SP-35, Pt. 1, 1963.
5. Danby, J. M. A.: Fundamentals of Celestial Mechanics. The MacMillan Co., c.1962.
6. Anon.: The American Ephemeris and Nautical Almanac for the Year 1964. Nautical Almanac Office, U.S. Naval Observatory, 1962.
7. Clohessy, W. H.; and Wiltshire, R. S.: Terminal Guidance System for Satellite Rendezvous. Jour. Aerospace Sci., vol. 27, no. 9, Sept. 1960, pp. 653-658, 674.
8. Anthony, Maurice L.; and Sasaki, Frank T.: The Rendezvous Problem for Nearly Circular Orbits. Paper No. 65-32, Am. Inst. Aeron. Astronaut., Jan. 1965.
9. Brouwer, Dirk; and Clemence, Gerald M.: Methods of Celestial Mechanics. Academic Press, Inc., 1961.
10. Dziobek, Otto: Mathematical Theories of Planetary Motions, Dover Publ., Inc., 1962.

TABLE I

COMPUTER PRINTOUT OF EXAMPLE 1

VELOCITY INCREMENT REQUIRED FOR RENDEZVOUS BETWEEN TWO ARBITRARY ELLIPTIC ORBITS

ET	EI	AI	OMEGAI	OMI	II	
0.93371999E-01	0.16724200E-01	0.65630100E 00	0.25387999E 03	0.23301999E 03	0.18499999E 01	
PHIO	NUO1	NUOL	MU	AT	TPER	VCT
0.32439999E 03	0.37000000E-00	0.37000000E-00	0.46789999E 22	0.74736999E 12	0.59348101E 08	0.79124083E 05
DTTI	DTTF	DTWI	DTWF		RMIN	H
0.16000000E 03	0.26000000E 03	0.	0.40000000E 02		0.39378100E-00	0.30000000E 01

NUO= 0.37
DELTTW= 0.

DELTTT	PHIF	NUI	E	A	I	THI	DELTTH	RHOM	DV1X	DV1Y	DV1Z	DV2X	DV2Y	DV2Z	DELV1	DELV2	DELV
160.0	423.41	0.37	0.6758	0.627	1.65	-134.33	296.14	0.20	0.78032	-0.44667	0.03413	-0.51167	-0.05421	0.01771	0.8998	0.5148	1.4146*
180.0	434.75	0.37	0.6661	0.596	1.87	-136.86	307.48	0.20	0.76728	-0.40968	0.02693	-0.44464	0.01959	0.01944	0.8702	0.4455	1.3157*
200.0	445.71	0.37	0.6883	0.583	2.24	-139.61	318.43	0.18	0.80293	-0.40036	0.01994	-0.44264	0.10123	0.02198	0.8974	0.4546	1.3520*
220.0	456.29	0.37	0.7463	0.581	2.88	-143.67	329.00	0.15	0.90147	-0.40946	0.01285	-0.51051	0.17498	0.02549	0.9902	0.5403	1.5305*
240.0	466.52	0.37	0.8392	0.586	4.19	-150.66	339.22	0.09	1.08507	-0.41653	0.00574	-0.65745	0.21178	0.02985	1.1623	0.6914	1.8537*
260.0	476.43	0.37	0.9427	0.598	7.86	-162.05	349.07	0.03	1.36525	-0.37671	-0.00035	-0.87692	0.15690	0.03412	1.4163	0.8915	2.3078*

DELTTW= 20.0

DELTTT	PHIF	NUI	E	A	I	THI	DELTTH	RHOM	DV1X	DV1Y	DV1Z	DV2X	DV2Y	DV2Z	DELV1	DELV2	DELV
160.0	434.75	20.74	0.6640	0.668	1.86	-129.46	287.10	0.22	0.87931	-0.19262	0.02853	-0.46880	-0.20692	0.02048	0.9006	0.5128	1.4135*
180.0	445.71	20.74	0.6432	0.630	2.01	-131.76	298.06	0.22	0.84048	-0.16181	0.02048	-0.40512	-0.12722	0.02168	0.8562	0.4252	1.2813*
200.0	456.29	20.74	0.6486	0.612	2.27	-134.02	308.63	0.22	0.84514	-0.14257	0.01297	-0.39640	-0.05278	0.02360	0.8572	0.4006	1.2578*
220.0	466.52	20.74	0.6817	0.606	2.70	-136.89	318.86	0.19	0.89663	-0.12717	0.00561	-0.43889	0.00647	0.02635	0.9056	0.4397	1.3453*
240.0	476.43	20.74	0.7448	0.607	3.43	-141.31	328.75	0.16	1.00350	-0.10339	-0.00183	-0.53243	0.03607	0.03004	1.0088	0.5345	1.5433*
260.0	486.05	20.74	0.8348	0.614	4.82	-148.51	338.34	0.10	1.17529	-0.04870	-0.00926	-0.67511	0.01106	0.03454	1.1763	0.6761	1.8524*

DELTTW= 40.0

DELTTT	PHIF	NUI	E	A	I	THI	DELTTH	RHOM	DV1X	DV1Y	DV1Z	DV2X	DV2Y	DV2Z	DELV1	DELV2	DELV
160.0	445.71	41.03	0.6567	0.716	1.86	-124.49	277.77	0.25	0.89817	0.07952	0.02058	-0.39029	-0.34103	0.02106	0.9019	0.5187	1.4206*
DELTTT	PHIF	NUI	E	A	I	THI	DELTTH	RHOM	DV1X	DV1Y	DV1Z	DV2X	DV2Y	DV2Z	DELV1	DELV2	DELV
180.0	456.29	41.03	0.6258	0.668	1.94	-126.59	288.35	0.25	0.83996	0.09298	0.01247	-0.33246	-0.25209	0.02159	0.8452	0.4178	1.2630*
200.0	466.52	41.03	0.6178	0.645	2.10	-128.52	298.57	0.25	0.81985	0.10684	0.00515	-0.32083	-0.17773	0.02273	0.8268	0.3675	1.1943*
220.0	476.43	41.03	0.6321	0.634	2.36	-130.68	308.47	0.23	0.83549	0.12600	-0.00178	-0.34789	-0.12324	0.02455	0.8449	0.3699	1.2148*
240.0	486.05	41.03	0.6699	0.632	2.76	-133.65	318.08	0.21	0.88813	0.15766	-0.00864	-0.40926	-0.09563	0.02712	0.9021	0.4212	1.3232*
260.0	495.41	41.03	0.7325	0.635	3.43	-138.19	327.43	0.17	0.98113	0.21375	-0.01563	-0.50157	-0.10622	0.03056	1.0043	0.5136	1.5179*

TABLE II
COMPUTER PRINTOUT OF EXAMPLE 2

VELOCITY INCREMENT REQUIRED FOR RENDEZVOUS BETWEEN TWO ARBITRARY ELLIPTIC ORBITS

ET	EI	AI	OMEGAI	OMI	II	
0.23400000E-01	0.29089106E-00	0.13543878E 01	0.	0.17574345E 01	0.	
PHIO	NUOI	NUOL	MU	AT	TPER	VCT
0.	0.48625083E-00	0.48625083E-00	0.14080000E 17	0.22480000E 08	0.56438179E 04	0.25026675E 05
DTTI	DTTF	DTWI	DTWF		RMIN	H
0.20000000E 02	0.40000000E 02	0.	0.30000000E 02		0.95400000E 00	0.30000000E 01
XO	YO	ZO	XODT	YODT	ZODT	
-0.16920000E-01	0.37599999E-01	0.	-0.37599999E-02	0.15260000E-00	0.	

NUO= 0.49

DELTTT= 20.0

DELTTW	PHIF	NUI	E	A	I	THI	DELTTH	RHOM	DV1X	DV1Y	DV1Z	DV2X	DV2Y	DV2Z	DELV1	DELV2	DELV
0.	79.17	0.49	0.0499	0.948	0.	107.97	76.93	0.96	0.05227	-0.14489	0.	-0.02179	0.03203	0.	0.1540	0.0387	0.1928
10.0	117.22	44.59	0.0735	0.964	0.	155.04	70.87	1.01	-0.01005	-0.20010	0.	-0.05237	0.05712	0.	0.2004	0.0775	0.2779
20.0	154.27	80.01	0.1979	1.271	0.	280.13	72.50	1.02	-0.05508	-0.43009	0.	0.01498	0.10739	0.	0.4336	0.1084	0.5420
30.0	190.85	106.69	0.6942	3.300	0.	292.34	82.40	1.01	0.09842	-0.74862	0.	0.08038	0.32180	0.	0.7551	0.3317	1.0867

DELTTT= 30.0

DELTTW	PHIF	NUI	E	A	I	THI	DELTTH	RHOM	DV1X	DV1Y	DV1Z	DV2X	DV2Y	DV2Z	DELV1	DELV2	DELV
0.	117.22	0.49	0.0379	0.974	0.	71.09	114.98	0.96	0.03930	-0.13126	0.	-0.02346	0.01591	0.	0.1370	0.0283	0.1654
10.0	154.27	44.59	0.0312	1.006	0.	132.66	107.92	1.02	-0.03276	-0.19063	0.	-0.03231	0.01876	0.	0.1934	0.0374	0.2308
20.0	190.85	80.01	0.1500	1.197	0.	266.85	109.08	1.02	-0.03019	-0.39549	0.	0.02658	0.08689	0.	0.3966	0.0909	0.4875
30.0	227.63	106.69	0.4355	1.713	0.	273.43	119.18	0.97	0.15056	-0.60254	-0.	0.01612	0.27834	0.	0.6211	0.2788	0.8999

DELTTT= 40.0

DELTTW	PHIF	NUI	E	A	I	THI	DELTTH	RHOM	DV1X	DV1Y	DV1Z	DV2X	DV2Y	DV2Z	DELV1	DELV2	DELV
0.	154.27	0.49	0.0367	0.987	0.	45.21	152.03	0.96	0.02890	-0.12482	0.	-0.02205	0.00286	0.	0.1281	0.0222	0.1504
10.0	190.85	44.59	0.0196	1.019	0.	113.84	144.49	1.02	-0.04107	-0.18982	0.	-0.01610	0.00656	0.	0.1942	0.0174	0.2116
20.0	227.63	80.01	0.1204	1.137	0.	245.09	145.86	1.00	-0.00549	-0.36346	-0.	0.00835	0.09499	0.	0.3635	0.0954	0.4589

DELTTW	PHIF	NUI	E	A	I	THI	DELTTH	RHOM	DV1X	DV1Y	DV1Z	DV2X	DV2Y	DV2Z	DELV1	DELV2	DELV
30.0	265.23	106.69	0.3124	1.344	0.	250.68	156.78	0.92	0.17237	-0.48852	-0.	-0.08178	0.24000	0.	0.5180	0.2535	0.7716*

TABLE III

COMPUTER PRINTOUT OF EXAMPLE 3

VELOCITY INCREMENT REQUIRED FOR RENDEZVOUS BETWEEN TWO ARBITRARY ELLIPTIC ORBITS

ET	EI	AI	OMEGAI	OMI	II	
0.50000000E 00	0.20000000E-00	0.90000000E 00	0.90000000E 02	-0.90000000E 02	0.30000000E 02	
PHIO	**NUOI**	**NUOL**	**MU**	**AT**	**TPER**	**VCT**
0.	0.	0.34000000E 03	0.09999999E 01	0.09999999E 01	0.62831853E 01	0.09999999E 01
DTTI	**DTTF**	**DTWI**	**DTWF**		**RMIN**	**H**
0.90000000E-01	0.10079999E 01	0.	0.		0.45000000E-00	0.30000000E 01

NUO= 0.
DELTTW= 0.

DELTTT	PHIF	NUI	E	A	I	THI	DELTTH	RHOM	DV1X	DV1Y	DV1Z	DV2X	DV2Y	DV2Z	DELV1	DELV2	DELV
0.0900	85.54	0.	1.0741	-6.373	-30.08	-42.83	86.14	0.62	-0.52733	0.28324	0.37539	0.15390	-0.08548	-0.78742	0.7066	0.8069	1.5134
0.0990	90.69	0.	0.9973	234.374	-30.00	-42.08	90.60	0.63	-0.52175	0.26365	0.29045	0.13805	-0.08853	-0.74180	0.6528	0.7597	1.4125
0.1080	95.41	0.	0.9331	9.475	-30.11	-41.27	94.69	0.63	-0.51804	0.24446	0.21509	0.12523	-0.09078	-0.70540	0.6119	0.7222	1.3340
0.1170	99.76	0.	0.8784	5.260	-30.36	-40.47	98.44	0.64	-0.51571	0.22560	0.14709	0.11451	-0.09247	-0.67604	0.5818	0.6919	1.2737
0.1260	103.76	0.	0.8311	3.818	-30.73	-39.65	101.91	0.64	-0.51449	0.20699	0.08487	0.10529	-0.09385	-0.65216	0.5610	0.6672	1.2283
0.1350	107.52	0.	0.7897	3.090	-31.19	-38.81	105.11	0.65	-0.51415	0.18856	0.02725	0.09715	-0.09508	-0.63265	0.5483	0.6471	1.1954
0.1440	110.99	0.	0.7531	2.651	-31.73	-37.95	108.08	0.65	-0.51458	0.17020	-0.02664	0.08982	-0.09633	-0.61667	0.5427	0.6306	1.1732
0.1530	114.25	0.	0.7265	2.358	-32.34	-37.07	110.83	0.66	-0.51566	0.15133	-0.07749	0.08308	-0.09769	-0.60359	0.5431	0.6171	1.1602
0.1620	117.30	0.	0.6912	2.148	-33.01	-36.17	113.41	0.66	-0.51732	0.13335	-0.12581	0.07681	-0.09927	-0.59294	0.5488	0.6061	1.1549
0.1710	120.18	0.	0.6647	1.990	-33.74	-35.24	115.81	0.67	-0.51949	0.11457	-0.17203	0.07089	-0.10113	-0.58433	0.5591	0.5972	1.1564
0.1800	122.90	0.	0.6407	1.868	-34.51	-34.28	118.06	0.67	-0.52213	0.09571	-0.21648	0.06528	-0.10333	-0.57746	0.5733	0.5903	1.1635
0.1890	125.48	0.	0.6188	1.770	-35.34	-33.31	120.17	0.67	-0.52519	0.07636	-0.25944	0.05990	-0.10592	-0.57209	0.5907	0.5849	1.1756
0.1980	127.93	0.	0.5987	1.690	-36.20	-32.30	122.16	0.68	-0.52864	0.05660	-0.30114	0.05474	-0.10893	-0.56803	0.6110	0.5810	1.1920
0.2070	130.27	0.	0.5802	1.624	-37.11	-31.27	124.04	0.68	-0.53244	0.03628	-0.34175	0.04976	-0.11238	-0.56511	0.6337	0.5783	1.2120
0.2160	132.50	0.	0.5632	1.568	-38.07	-30.22	125.81	0.68	-0.53655	0.01534	-0.38143	0.04496	-0.11632	-0.56321	0.6585	0.5768	1.2353

"The aeronautical and space activities of the United States shall be conducted so as to contribute . . . to the expansion of human knowledge of phenomena in the atmosphere and space. The Administration shall provide for the widest practicable and appropriate dissemination of information concerning its activities and the results thereof."

—NATIONAL AERONAUTICS AND SPACE ACT OF 1958

NASA SCIENTIFIC AND TECHNICAL PUBLICATIONS

TECHNICAL REPORTS: Scientific and technical information considered important, complete, and a lasting contribution to existing knowledge.

TECHNICAL NOTES: Information less broad in scope but nevertheless of importance as a contribution to existing knowledge.

TECHNICAL MEMORANDUMS: Information receiving limited distribution because of preliminary data, security classification, or other reasons.

CONTRACTOR REPORTS: Technical information generated in connection with a NASA contract or grant and released under NASA auspices.

TECHNICAL TRANSLATIONS: Information published in a foreign language considered to merit NASA distribution in English.

TECHNICAL REPRINTS: Information derived from NASA activities and initially published in the form of journal articles.

SPECIAL PUBLICATIONS: Information derived from or of value to NASA activities but not necessarily reporting the results of individual NASA-programmed scientific efforts. Publications include conference proceedings, monographs, data compilations, handbooks, sourcebooks, and special bibliographies.

Details on the availability of these publications may be obtained from:

SCIENTIFIC AND TECHNICAL INFORMATION DIVISION

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

Washington, D.C. 20546