X-551-71-245
PREPRINT

NASA TM X- 65633

A COMPARISON OF AN ANALYTICAL AND NUMERICAL SOLUTION OF THE RELATIVE MOTION OF TWO CLOSE SATELLITES OF AN OBLATE PLANET

RICHARD W. BARBIERI
JOEL J. MASHBAUM

JUNE 1971

GSFC

GODDARD SPACE FLIGHT CENTER
GREENBELT, MARYLAND

X-551-71-245

A COMPARISON OF AN ANALYTICAL AND NUMERICAL SOLUTION OF THE RELATIVE MOTION OF TWO CLOSE SATELLITES OF AN OBLATE PLANET

Richard W. Barbieri
Joel J. Mashbaum

June 1971

Goddard Space Flight Center
Greenbelt, Maryland

TABLE OF CONTENTS

A COMPARISON OF AN ANALYTICAL AND NUMERICAL SOLUTION OF THE RELATIVE MOTION OF TWO CLOSE SATELLITES OF AN OBLATE PLANET

Richard W. Barbieri
Joel J. Mashbaum

ABSTRACT

This report presents the results of a study undertaken to compare an analytical theory of relative motion, developed by R. Barbieri (Ref. 1), with a numerical integration of the equations of planar relative motion.

The equations of relative motion were derived from a Lagrangian formulation and is presented in detail in [1]. The numerical integration has been carried out with a program that integrates a system of ordinary differential equations which may or may not be coupled. On the other hand, the analytical solution has been constructed successively in terms of powers of the eccentricity of the reference satellite and powers of a small parameter. Since for the application to problems of orbiting long baseline interferometry the eccentricity is small ($\leq .01$), the expansion of the solution in powers of eccentricity is carried to the first power only.

The results indicate, as expected, good agreement, using small initial velocities, over several days for a range of eccentricities. For example, at a semi-major axis of 17,000 km, and eccentricity of .01 and initial velocity components of .01 km/sec the analytical development yields a baseline distance which differs from the distance as calculated numerically by no more than 5% over the first 3 1/2 days. On the other hand when the semi-major is increased to 20,000 km, the eccentricity decreased to .0001 the baseline distance as calculated by both programs differ by no more than 5% over the first 6 days. Comparisons are presented to show the effect of including the oblateness of the central body. Results are documented to indicate the dependence of the motion of one satellite about the other upon a variety of initial conditions.

PRECEDING PAGE BLANK NOT FILMED

GLOSSARY OF SYMBOLS

$\vec{r}_m$ radius vector of the M satellite with respect to the center of mass of the central body

$\vec{r}_{0'}$ radius vector of the O' satellite with respect to the center of mass of the central body;

$\vec{R}$ position vector of M with respect to O'; has components $\{x, y, z\}$

a the semi-major axis of the orbit of O'

e the eccentricity of the orbit of O'

ι the inclination of the orbit of O'

Ω the longitude of the ascending node of the orbit of O'

w the argument of perigee of O'

w_0 the secular rate of change of the argument of perigee

v the true anomaly of O'

u the sum $(w + v)$

ω the angular velocity of O'; has components $\{\omega_x, \omega_y, \omega_z\}$

$\{\dot{x}_{0'}, \dot{y}_{0'}, \dot{z}_{0'}\}$ the velocity components of O' in the rotating coordinate system

$\{F_x, F_y, F_z\}$ the components of force, in the rotating coordinate system, acting on the M satellite.

μ the gravitational constant of the central body

R_e the equatorial radius of the central body

J_{20} the first non zero coefficient in the spherical harmonic expansion of the gravity field of the central body

U the potential energy of the M satellite

ϵ a small parameter $\left(= \frac{3}{2} J_{20}\right)$

$n = \sqrt{\mu a^{-3}}$

$\bar{n} = n + \dot{\Omega} \cos \iota$

$\tilde{\omega}_z = \frac{1}{8} \left(\frac{R_e}{a}\right)^2 \bar{n} \left(1 - \frac{3}{2} \sin^2 \iota\right)$

R_A the analytical baseline length

R_N the numerical baseline length

A COMPARISON OF AN ANALYTICAL AND NUMERICAL SOLUTION OF THE RELATIVE MOTION OF TWO CLOSE SATELLITES OF AN OBLATE PLANET

INTRODUCTION

In Reference [1] a solution of the equations of planar relative motion was constructed taking into account the oblateness of the central body. Since the publication of those results an effort has been made to determine how the analytical solution compares to the numerical integration of the equations of planar relative motion. This report represents a culmination of that effort.

A study of the relative motion of two close spacecraft orbiting an oblate planet has some rather interesting applications. For example the concept of an orbiting long baseline interferometer has some intriguing implications. Not only is it possible to get data unperturbed by the troposphere or ionosphere from various radio sources but it is conceivable that such a system, yielding extremely accurate positions of radio sources, can be used to provide an independent estimation of the structure of the troposphere and ionosphere and consequently an independent verification of refraction effects.

The analytical representation was constructed by expanding the equations of motion into powers of eccentricity and a small parameter, ϵ, which is related to the second zonal harmonic coefficient J_2. The equations of motion resulting from this operation are then solved by successive approximations. The first step in this successive approximation technique yields an almost periodic solution, independent of ϵ and eccentricity, whose frequency depends upon the modified mean motion. The second step introduces the dependence of the motion upon the eccentricity of the reference satellite and also yields an almost periodic solution with three frequencies appearing. The last step of the procedure accounts for the oblateness of the central body and yields a solution which depends upon four frequencies, three of which are dependent upon the secular rate of increase of the argument of perigee. Furthermore the last step of the procedure also introduces a purely secular term; mixed secular terms have been eliminated by making the angular velocity of the reference satellite dependent upon the small parameter.

The routine FNOL2 uses the fourth order Runge-Kutta and/or Adams-Moulton methods of solution of ordinary differential equations which may or may not be coupled. As currently modified, the program uses a double precision arithmetic throughout. Furthermore the truncation error can be held within input bounds by giving the user an option of automatically varying the step size. More detailed information is available in [2].

A description of the subroutines used in the numerical and analytical programs is presented in Appendix II and Appendix III respectively. A listing of the subroutines is given in Appendix IV and Appendix V.

THE EQUATIONS OF RELATIVE MOTION

The equations of relative motion which are of main concern in this report are derivable from the following system of coupled differential equations:

$$\ddot{x} + \ddot{x}_{0'} - 2\,\dot{y}\,\omega_z - y\,\dot{\omega}_z - \dot{y}_{0'}\,\omega_z + \omega_z\,(z\,\omega_x - x\,\omega_z) + \frac{1}{m}\,\frac{\partial U}{\partial x} = F_x \tag{1}$$

$$\ddot{y} + \ddot{y}_{0'} + 2\,(\dot{x}\,\omega_z - \dot{z}\,\omega_x) + x\,\dot{\omega}_z - z\,\dot{\omega}_x + \dot{x}_{0'}\,\omega_z - \dot{z}_{0'}\,\omega_x - y\,\omega_x^2 - y\,\omega_z^2 + \frac{1}{m}\,\frac{\partial U}{\partial y} = F_y \tag{2}$$

$$\ddot{z} + \ddot{z}_{0'} + 2\,\dot{y}\,\omega_x + y\,\dot{\omega}_x + \dot{y}_{0'}\,\omega_x - \omega_x\,(z\,\omega_x - x\,\omega_z) + \frac{1}{m}\,\frac{\partial U}{\partial z} = F_z. \tag{3}$$

See Figure 1 for the geometry.

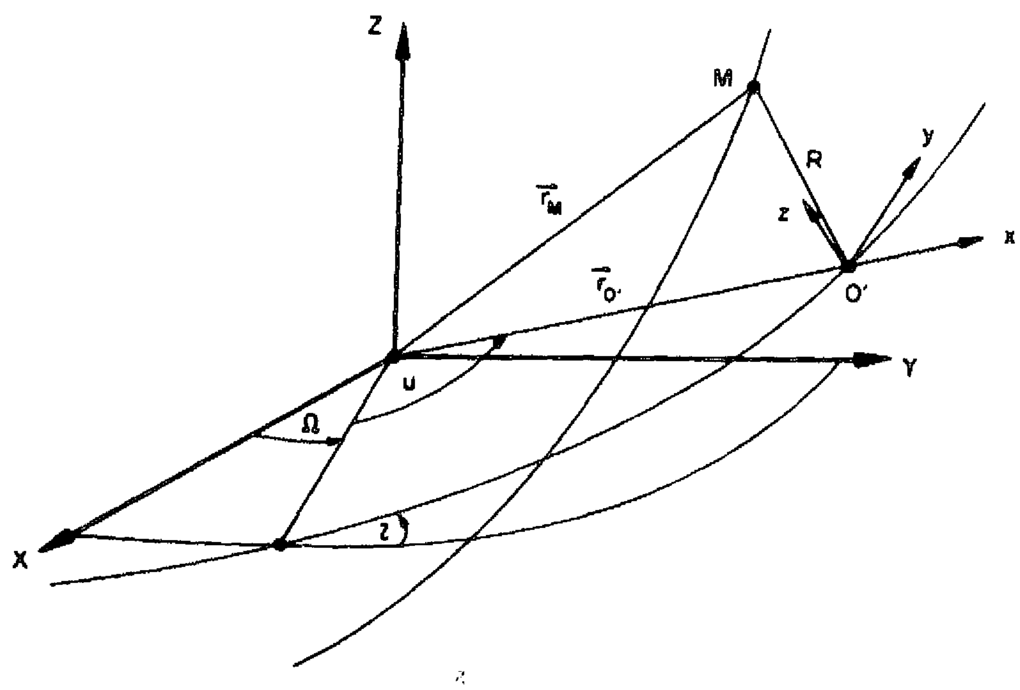


Figure 1. Geometry

A solution to this system of equations presents a formidable task since they are coupled. In Reference [1] the angular velocity component ω_x is neglected thereby uncoupling the motion; this imposes a constraint on the rotation of the orbital plane of O' about $\vec{r}_1$. In particular the motion governed by the differential equation $\dot{\iota} \cos u + \dot{\Omega} \sin u \sin \iota$ is being neglected. Making the substitutions

$$U = - \frac{\mu}{r_m} \tag{4}$$

$$F_x = \frac{3}{2} \mu J_2 R_e^2 \frac{x + x_{0'}}{x_{0'}^5} (1 - 3 \sin^2 \iota \sin^2 u) \tag{5}$$

$$F_y = \frac{3}{2} \mu J_2 R_e^2 \frac{\sin u \cos u \sin^2 \iota}{x_{0'}^4} \tag{6}$$

$$\ddot{x}_{0'} - x_{0'} \omega_z^2 + \frac{\mu}{x_{0'}^2} = (F_x)_{0'} \tag{7}$$

$$\frac{1}{x_{0'}} \frac{d}{dt} (x_{0'}^2 \omega_z) = (F_y)_{0'}, \tag{8}$$

reduces the equations of planar relative motion to the form

$$\ddot{x} - 2 \dot{y} \omega_z - y \dot{\omega}_z - x \omega_z^2 + \frac{\mu}{x_{0'}^3} x = \frac{3}{2} \mu J_2 R_e^2 \frac{x}{x_{0'}^5} (1 - 3 \sin^2 \iota \sin^2 u) \tag{9}$$

$$\ddot{y} + 2 \dot{x} \omega_z + x \dot{\omega}_z - y \omega_z^2 + \frac{\mu}{x_{0'}^3} y = \frac{3}{2} \mu J_2 R_e^2 \frac{y}{x_{0'}^5} \sin^2 \iota \cos 2u \tag{10}$$

These represent the equations which have been programmed for numerical solution by FNOL2.

In order to present an analytical solution of these equations certain transformations and representations must be used so as to have the time appearing explicitly. To begin, $\dot{\omega}_z$ is transformed to

$$- 2 \frac{x_{0'} \omega_z^2}{a (1 - e^2)} e \sin v$$

and then the true anomaly of the reference satellite is introduced via the transformation

$$x_{0'} = \frac{a (1 - e^2)}{1 + e \cos v} . \tag{11}$$

Finally the true anomaly is expressed as a function of the mean anomaly through the representation

$$\left(\frac{r}{a}\right)^{\alpha} \exp (i \gamma v) = \exp i \gamma M - \frac{e}{2} (\alpha - 2 \gamma) \exp i (\gamma + 1) M - \frac{e}{2} (\alpha + 2 \gamma) \exp i (\gamma - 1) M \tag{12}$$

where

$$\alpha, \gamma = 0, \pm 1, \pm 2, \ldots$$

Making the necessary substitutions leads to the following system of equations which are more amenable to analytical solution

$$\ddot{x} - 2 \dot{y} \omega_z + 2 y \omega_z^2 \frac{e \sin v}{1 + e \cos v} - x \omega_z^2 \frac{e \cos v}{1 + e \cos v} = \epsilon \left(\frac{R_e}{a}\right)^2 x (1 - e^2)^{-2} \omega_z^2 (1 + e \cos v) (1 - 3 \sin^2 \iota \sin^2 u) \tag{13}$$

$$\ddot{y} + 2\,\dot{x}\,\omega_z - 2\,x\,\omega_z^2\,\frac{e\sin v}{1 + \cos v} - y\,\omega_z^2\,\frac{e\cos v}{1 + e\cos v}$$

$$= \epsilon\left(\frac{R_e}{a}\right)^2 y\,(1 - e^2)^{-2}\,\omega_z^2\,(1 + e\cos v)\cos 2u\,\sin^2 \iota. \tag{14}$$

These equations, as has been mentioned, are solved using the method of successive approximations starting with a representation of the solution in the form

$$x\,(t) = x_{00}\,(t) + e\,x_{01}\,(t) + \epsilon\,x_{10}\,(t) \tag{15}$$

$$y\,(t) = y_{00}\,(t) + e\,y_{01}\,(t) + \epsilon\,y_{10}\,(t) \tag{16}$$

Substituting these two expressions into (13) and (14), using (12) to get the mean anomaly and expressing ω_z in the form

$$\omega_z = n + \dot{\Omega}\cos \iota + 2\,n\,e\cos v + \epsilon\,\tilde{\omega}_z \tag{17}$$

yields the system of equations which govern the first order behavior of the satellite:

$$\ddot{x}_{00} - 2\,\bar{n}\,\dot{y}_{00} = 0$$

$$\ddot{x}_{01} - 2\,\bar{n}\,\dot{y}_{01} - 4\,n\,\dot{y}_{00}\cos n\,t + 2\,\bar{n}^2\,y_{00}\sin n\,t \tag{18}$$

$$- \bar{n}^2\,x_{00}\cos n\,t = 0$$

$$\ddot{x}_{10} - 2\,\bar{n}\,\dot{y}_{10} - 2\,\tilde{\omega}_z\,\dot{y}_{00} = \left(\frac{R_e}{a}\right)^2 \bar{n}^2\,x_{00}\left(1 - \frac{3}{2}\sin^2 \iota\right)$$

$$+ \frac{3}{2}\left(\frac{R_e}{a}\right)^2 \bar{n}^2\,x_{00}\sin^2 \iota\,\cos\,(2\,w_0 + 2\,n)\,t$$

$$\ddot{y}_{00} + 2\,\bar{n}\,\dot{x}_{00} = 0$$

$$\begin{aligned} &\ddot{y}_{01} + 2\,\bar{n}\,\dot{x}_{01} + 4\,n\,\dot{x}_{00}\cos n\,t - 2\,\bar{n}^2\,x_{00}\sin n\,t \\ &\quad - \bar{n}^2\,y_{00}\cos n\,t = 0 \end{aligned} \tag{19}$$

$$\ddot{y}_{10} + 2\,\bar{n}\,\dot{x}_{10} + 2\,\tilde{\omega}_z\,\dot{x}_{00} = \left(\frac{R_e}{a}\right)^2 \bar{n}^2\,y_{00}\sin^2 i\,\cos\,(2\,w_0 + 2\,n)\,t.$$

where the expressions M = nt and w = w_0 t have been used. The solution of the system (18) - (19) is presented in [1]; the results are that

$$x_{00}\,(t) = \alpha_1\sin 2\,\bar{n}\,t + \alpha_2\cos 2\,\bar{n}\,t - \frac{\alpha_0}{2\bar{n}} \tag{20}$$

$$y_{00}\,(t) = \alpha_1\cos 2\,\bar{n}\,t - \alpha_2\sin 2\,\bar{n}\,t + \alpha_3 \tag{21}$$

$$\begin{aligned} x_{01}\,(t) = {} & \beta^{01}_{1x}\sin n\,t + \beta^{01}_{2x}\cos n\,t + \beta^{01}_{3x}\sin\,(2\,\bar{n} + n)\,t \\ & + \beta^{01}_{4x}\sin\,(2\,\bar{n} - n)\,t + \beta^{01}_{5x}\cos\,(2\,\bar{n} + n)\,t \\ & + \beta^{01}_{6x}\cos\,(2\,\bar{n} - n)\,t \end{aligned} \tag{22}$$

$$\begin{aligned} y_{01}\,(t) = {} & \beta^{01}_{1y}\sin n\,t + \beta^{01}_{2y}\cos n\,t + \beta^{01}_{3y}\sin\,(2\,\bar{n} + n)\,t \\ & + \beta^{01}_{4y}\sin\,(2\,\bar{n} - n)\,t + \beta^{01}_{5y}\cos\,(2\,\bar{n} + n)\,t \\ & + \beta^{01}_{6y}\cos\,(2\,\bar{n} - n)\,t \end{aligned} \tag{23}$$

$$\begin{aligned} x_{10}\,(t) = {} & \beta^{10}_{1x}\sin 2\,(w_0 + n)\,t + \beta^{10}_{2x}\cos 2\,(w_0 + n)\,t \\ & + \beta^{10}_{3x}\sin 2\,(w_0 + n + \bar{n})\,t + \beta^{10}_{4x}\sin 2\,(w_0 + n - \bar{n})\,t \end{aligned} \tag{24}$$

$$+ \beta^{10}_{5x} \cos 2 (w_0 + n + \bar{n})\, t + \beta^{10}_{6x} \cos 2 (w_0 + n - \bar{n})\, t$$

$$+ \beta^{10}_{7x}.$$

$$\begin{aligned} y_{10}(t) = {} & \beta^{10}_{1y} \sin 2 \bar{n}\, t + \beta^{10}_{2y} \cos 2 \bar{n}\, t + \beta^{10}_{3y} \sin 2 (w_0 + n)\, t \\ & + \beta^{10}_{4y} \cos 2 (w_0 + n)\, t + \beta^{10}_{5y} \sin 2 (w_0 + n + \bar{n})\, t \\ & + \beta^{10}_{6y} \sin 2 (w_0 + n - \bar{n})\, t + \beta^{10}_{7y} \cos 2 (w_0 + n + \bar{n})\, t \\ & + \beta^{10}_{8y} \cos 2 (w_0 + n - \bar{n})\, t + \beta^{10}_{9y}\, t. \end{aligned} \tag{25}$$

The functions α_j with $j = 0, 1, 2, 3$, $\{\beta^{01}_{jx}, \beta^{01}_{iy}\}$ with $j = 1, 2, \ldots, 6$ and $\{\beta^{10}_{jx}, \beta^{10}_{jy}\}$ with $j = 1, 2, \ldots, 7$ are given explicitly in the Appendix I.

Turning attention to the behavior of the motion of the reference satellite, it will be noticed that in the numerical integration program this behavior is specified by a modification of the planetary equations of Lagrange to allow for very small eccentricities. To be specific, the Lagrange equations, before any modifications, take the form

$$\dot{a} = \frac{2}{n \sqrt{1 - e^2}} \{S\, e \sin v + T (1 + e \cos v)\}$$

$$\dot{e} = \frac{\sqrt{1 - e^2}}{n\, a} \left\{ S \sin v + T \left[e \frac{x_0'}{p} + \left(1 + \frac{x_0'}{p}\right) \cos v \right] \right\} \tag{26}$$

$$\dot{\iota} = \frac{x_0'}{n\, a^2 \sqrt{1 - e^2}} W \cos (v + w)$$

$$\dot{\Omega} = \dot{\iota}\, \frac{\tan (v + w)}{\sin \iota}$$

$$\dot{\omega} = \frac{\sqrt{1 - e^2}}{n\,a\,e} \left\{ - S \cos v + T \left(1 + \frac{x_0{}'}{p} \right) \sin v \right\} - \dot{\Omega} \cos \iota$$

$$\dot{v} = \frac{n\,a^2 \sqrt{1 - e^2}}{x_0^2{}'} + \frac{1}{e} \; \frac{\sqrt{1 - e^2}}{n\,a} \left[S \cos v - T \left(1 + \frac{x_0{}'}{p} \right) \sin v \right]$$

where S is the force due to the first zonal harmonic directed along the radius vector to O', T is the force due to the first zonal harmonic 90° ahead of S and located in the orbit plane of O', W completes the right handed coordinate system.

Because of the presence of the eccentricity in the denominator of the $\dot{\omega}$ and $\dot{v}$ equations, a study of the motion for very small eccentricities becomes quite difficult. To overcome this situtation a transformation of variables is made; in particular let,

$$\ell = e \cos w$$

$$m = e \sin w$$

and replace the equation for $\dot{v}$ by the equation for $\dot{u}$ where $u = v + w$. These transformations lead to the following system of equations describing the motion of the reference satellite:

$$\dot{a} = - 3 \frac{\mu J_2 R_e^2}{n \sqrt{1 - \ell^2 - m^2}} \left\{ (\ell \sin u - m \cos u) \frac{1 - 3 \sin^2 \iota \sin^2 u}{x_0^4{}'} \right.$$

$$\left. + (1 + \ell \cos u + m \sin u) \frac{\sin^2 \iota \sin 2u}{x_0^4{}'} \right\}$$

$$\dot{\iota} = - \frac{3}{2} \frac{\mu J_2 R_e^2}{n a^2 \sqrt{1 - \ell^2 - m^2}} \; \frac{\sin 2\iota \sin u \cos u}{x_0^3{}'}$$

$$\dot{\Omega} = i \frac{\tan u}{\sin \iota}$$

$$\dot{\ell} = m \dot{\Omega} \cos \iota - \frac{3}{2} \mu J_2 R_e^2 \frac{\sqrt{1 - \ell^2 - m^2}}{n a} \left\{ \frac{(1 - 3 \sin^2 \iota \sin^2 u) \sin u}{x_0^{4\prime}} \right.$$

$$\left. - \left[\left(\frac{x_0'}{p} \right) \ell + \left(1 + \frac{x_0'}{p} \right) \cos u \right] \frac{\sin^2 \iota \sin 2u}{x_0^{4\prime}} \right\} \tag{27}$$

$$\dot{m} = - \ell \dot{\Omega} \cos \iota + \frac{3}{2} \mu J_2 R_e^2 \frac{\sqrt{1 - \ell^2 - m^2}}{n a} \left\{ \frac{(1 - 3 \sin^2 \iota \sin^2 u) \cos u}{x_0^{4\prime}} \right.$$

$$\left. - \left[\left(\frac{x_0'}{p} \right) m + \left(1 + \frac{x_0'}{p} \right) \sin u \right] \frac{\sin^2 \iota \sin 2u}{x_0^{4\prime}} \right\}$$

$$\dot{u} = \frac{n a^2 \sqrt{1 - \ell^2 - m^2}}{x_0^{2\prime}} - \dot{\Omega} \cos \iota$$

where

$$x_0' = \frac{a (1 - \ell^2 - m^2)}{1 + \ell \cos u + m \sin u}$$

On the other hand, the analytical development uses a slightly different set of equations to specify the position of the reference satellite; for example, the transformation to ℓ and m variables is not performed because the secular rate of the argument of perifocus is required and, as will be shown, this parameter does not contain the eccentricity as a small divisor.

The secular rate of any of the orbital elements can be obtained quite simply with the help of the Hansen coefficients. From Tisserand [3] a discussion of the development of the functions

$$\left(\frac{r}{a}\right)^{\alpha} \sin \gamma v, \; \left(\frac{r}{a}\right)^{\alpha} \cos \gamma v$$

in terms of the mean anomaly, M, is presented. The Fourier development of $(r/a)^{\alpha} \exp i \gamma v$ is given as

$$\left(\frac{r}{a}\right)^{\alpha} \exp i \gamma v = \sum_{k=-\infty}^{\infty} X_K^{\alpha,\gamma} \exp i k M \tag{28}$$

where the development shows that the coefficient independent of M (and hence independent of time explicitly) is given by

$$X_0^{\alpha,\gamma} = (-1)^{\gamma} \frac{(\alpha+2)(\alpha+3)\cdots(\alpha+\gamma+1)}{\gamma!} \left(\frac{e}{2}\right)^{\gamma} F\left(\frac{\gamma-\alpha-1}{2}, \frac{\gamma-\alpha}{2}, 1+\gamma, e^2\right); \tag{29}$$

that is, $X_0^{\alpha,\gamma}$ is the constant coefficient in the Fourier development; the dependence on the eccentricity is explicitly shown in the expansion $X_0^{\alpha,\gamma}$.

Carrying out these operations in the $\dot{\omega}$ equation of (26) leads to an equation for the secular variation of $\dot{w}$, namely $\dot{w}_0$, given by

$$\dot{w}_0 = \frac{3}{4} n J_2 \frac{R_e^2}{p^2} (5 \cos^2 \iota - 1). \tag{30}$$

The remaining set of equations governing the motion of the reference satellite are given by

$$\dot{a} = -3 \mu J_2 \frac{R_e^2}{x_0^4,} \frac{1}{n\sqrt{1-e^2}} \Bigg\{(1 - 3 \sin^2 \iota \sin^2 u) e \sin (u - w_0)$$

$$+ \frac{p}{x_0,} \sin^2 \iota \sin 2u\Bigg\}$$

$$\dot{e} = -\frac{3}{2}\mu J_2 \frac{R_e^2}{x_0^4{}'} \frac{\sqrt{1-e^2}}{n\,a} (1 - 3\sin^2\iota\,\sin^2 u)\sin(u - w_0)$$

$$+\frac{x_0{}'}{p}\left[e\cos^2(u - w_0) + 2\cos(u - w_0) + e\right]\sin^2\iota\,\sin 2u$$

$$\dot{\iota} = -\frac{3}{4}\frac{\mu J_2 R_e^2}{n\,a^2\sqrt{1-e^2}} \frac{\sin 2\iota\,\sin 2u}{x_0^3{}'} \tag{31}$$

$$\dot{\Omega} = \dot{\iota}\,\frac{\tan u}{\sin\iota}$$

$$\dot{u} = \frac{n\,a^2\sqrt{1-e^2}}{x_0^2{}'} - \dot{\Omega}\cos$$

where

$$x_0{}' = \frac{a(1-e^2)}{1 + e\cos(u - w_0)}$$

THE RESULTS

What will now be compared are the results generated by (20) thru (25), with the results of the numerical integration of (9) and (10). The purpose of this comparison is to determine the interval of time over which the analytical solution is valid and to illustrate the dependence of the analytical development on small eccentricities. Furthermore it will be shown just how critical the initial relative velocity is to the motion.

Comparisons have been obtained for about 25 cases where the semi-major axis takes on the values 17,000 km and 20,000 km, the eccentricity varies between .0001 and .01 and the initial relative velocity varies between 1 m/sec to

10 m/sec. All cases have the inclination of the reference satellite initially at 10°, the longitude of the ascending node at 20°, the argument of perigee at 0.°0, and the true anomaly initially at 45°.

Tables I and II were generated for a semi-major axis of 17,000 km and eccentricity of .01; the only parameter which was varied was the initial velocity. It is noticed that an increase of one order of magnitude in the velocity increases the baseline magnitude by a factor of roughly 10; coupled with this increase is a significant difference in the orientation of the baseline.

Table III differs from Table I in that only the eccentricity is different. As expected, the difference in the magnitude and orientation of the baseline is negligible.

Tables IV and V were generated with a semi-major axis of 20,000 km. and a nearly circular orbit; once again only the initial velocity was changed. It is noticed immediately that the baseline undergoes slightly larger excursions when compared to the cases where a = 17,000 km; furthermore the orientation of the baseline is much different due to the presence of n and $\overline{n}$ in the equations (20) thru (25). The agreement between the analytical and numerical is much better than when the semi-major axis is 17,000 km; in fact not only is the agreement better but it lasts longer. For example in Table I it is noticed that after about 5 days the agreement between the x components begins to break down.

Table VI shows what can happen with a slight variation in the initial velocity.

Finally the graphs I through IV are included to show the accumulative effect of the oblateness of the central body, in these cases the earth. What is shown is the variation of the difference in baseline length versus time. R_A and R_N are the baseline lengths as computed in the analytical program and numerical program respectively. The insensitivity of the motion as a function of eccentricity is clearly shown as is the strong dependence of the motion on the semi-major axis of the reference satellite.

COMMENTS, CONCLUSIONS AND SUMMARY

What has been presented are some results generated by comparing an analytical development of the equations of relative motion to a numerical solution of these equations. The results clearly indicate the motion is strongly dependent upon the semi-major axis of the reference satellite and on the initial relative velocity. For example the following summary shows the dependence on the aforementioned parameters and indicates the percent difference between the analytical value of the baseline distance and the numerical value; also shown is the interval of time the two values agree to within 5 percent.

a (km)	e	$\dot{x}_0$ km/s	$\dot{y}_0$ km/s	Time Interval	% Difference of Baseline Length
17,000	.01	.01	.01	3 1/2 Days	< 5
17,000	.01	.001	.001	3 1/2 Days	< 5
17,000	.001	.01	.01	3 1/2 Days	< 5
20,000	.0001	.01	.01	6 Days	< 5
20,000	.0001	.001	.001	6 Days	< 5

Qualitative and quantitative information has been presented to indicate the accumulative effects of oblateness upon the relative motion; it is clear from the cases shown that a variation of the eccentricity from 0.0001 to 0.01 has little effect on the relative motion over a period of 7 days. For example the following table presents the difference between the numerical and analytical values for the baseline length at the 7th day; the initial components of velocity are .001 km/s.

a (km)	e	$(R_A - R_N)$ km	J_2
17,000	.0001	- .1538	off
17,000	.0001	+ .167	on
17,000	.001	- .1680	off
17,000	.001	+ .151	on
20,000	.0001	- .0114	off
20,000	.0001	- .312	on
20,000	.001	+ .0063	off
20,000	.001	- .2932	on

A time history over 7 days of the difference $R_A - R_N$ is shown in the graphs.

Agreement of the analytical solution with the numerical is much better at larger semi-major axes since the periodic behavior of Ω and ω is less pronounced at the semi-major axes of 20,000 km or more.

It must be mentioned that the analytical development was carried out under the assumption that n and $\bar{n}$ are constant. Being rigorous, this assumption is not quite valid; the following two tables show how n, $\bar{n}$ and the coefficients of integration vary over an interval of seven days.

$$a = 20{,}000 \text{ km}, \quad e = .001 \quad J_2 \neq 0$$

$$.2232143 \times 10^{-3}/\text{sec} \leq n \leq .2232177 \times 10^{-3}/\text{sec}$$

$$.2231463 \times 10^{-3}/\text{sec} \leq \bar{n} \leq .2232145 \times 10^{-3}/\text{sec}$$

$$.143879 \times 10^{-2} \text{ km/sec} \leq \alpha_0 \leq .144600 \times 10^{-2} \text{ km/sec}$$

$$2.233363 \text{ km} \leq \alpha_1 \leq 2.238736 \text{ km}$$

$$-2.237675 \text{ km} \leq \alpha_2 \leq -2.238756 \text{ km}$$

$$-1.236651 \text{ km} \leq \alpha_3 \leq -1.237939 \text{ km}$$

$$a = 17{,}000 \quad e = .001 \quad J_2 \neq 0$$

$$.2848343 \times 10^{-3}/\text{sec} \leq n \leq .2848404 \times 10^{-3}/\text{sec}$$

$$.2847141 \times 10^{-3}/\text{sec} \leq \bar{n} \leq .2848344 \times 10^{-3}/\text{sec}$$

$$.1569019 \times 10^{-2} \text{ km/sec} \leq \alpha_0 \leq .1569368 \times 10^{-2} \text{ km/sec}$$

$$1.753537 \text{ km} \leq \alpha_1 \leq 1.754718 \text{ km}$$

$$-1.753565 \text{ km} \leq \alpha_2 \leq -1.754745 \text{ km}$$

$$-0.7525190 \text{ km} \leq \alpha_3 \leq -0.7539899 \text{ km}.$$

Comparisons of the use of w in the numerical program versus the use of w_0 in the analytical program indicate that for a semi-major axis of 17,000 km the analytical value of $u = v + w_0$ differs from the numerical value of $u = v + w$ by about one to three degrees. The parameter w_0 was used in the analytical development to avoid consideration of differential equations with almost periodic coefficients which arise when the oblateness is included. The agreement shown here between the analytical and numerical programs indicates that the former would serve as an excellent tool for very early orbit determination of the relative motion of two close satellites. For the type studies alluded to in the introduction, namely, the study of spectra from radio sources and, the study of continental drift, a very precise orbit determination program is required. This means including the perturbation effects due to higher order harmonics and the presence of the sun and moon. Naturally then, an analytic solution of the form presented in [1] would be quite cumbersome and time consuming to carry out.

ACKNOWLEDGEMENT

The first author thanks Dr. Peter Musen and Mr. Eugene Lefferts for some helpful discussions during the course of this work.

REFERENCES

[1] Barbieri, R. W., "An Analytical Development of the Relative Motion of Two Close Satellites of an Oblate Planet," Goddard Space Flight Center, X-551-70-313, Aug. 1970.

[2] Belliveau, L. J. and Linnekin, J., "FNOL2, A Fortran Subroutine for the Solution of Ordinary Differential Equations with Automatic Adjustment of the Interval of Integration," U. S. Naval Ordnance Laboratory, NOLTR 63-171.

[3] Tisserand, F., "Méchanique Céleste," Vol. I, Chapter 15.

TABLE I

$a = 17{,}000$ $\quad x_0 = 1.0$ km $\quad \dot{x}_0 = .01$ km/s $\quad J_2 \neq 0$

$e = .01$ $\quad y_0 = 1.0$ km $\quad \dot{y}_0 = .01$ km/s

Time (hrs.)	x (km)		y (km)		R (km)	
	Analytical	Numerical	Analytical	Numerical	Analytical	Numerical
12	-5.3791	-5.6772	-10.222	-9.9053	11.551	11.415
24	-5.1736	-6.0541	-23.046	-22.872	23.619	23.660
36	1.4709	.1904	-34.015	-34.464	34.046	34.465
48	12.812	11.552	-40.184	-41.484	42.178	43.062
60	25.854	25.078	-39.748	-41.897	47.417	48.829
72	36.934	37.148	-32.622	-35.429	49.278	51.335
84	42.745	44.465	-20.745	-23.675	47.513	50.375
96	41.555	44.957	-7.6492	-9.6819	42.253	45.988
108	33.913	38.346	2.8116	2.8599	34.030	38.452
120	22.307	26.221	7.8796	10.594	23.658	28.281
132	10.026	11.633	6.6392	11.377	12.025	16.272
144	.04627	-1.6942	-.15822	4.8340	.1648	5.1223
156	-5.5229	-10.306	-10.720	-7.5457	12.059	12.773
168	-5.5038	-11.886	-22.719	-22.760	23.376	25.677

TABLE II

$a = 17{,}000$ km $x_0 = 1.0$ km $\dot{x}_0 = .001$ km/s

$e = .01$ $y_0 = 1.0$ km $\dot{y}_0 = .001$ km/s $J_2 \neq 0$

Time (hrs.)	x (km)		y (km)		R (km)	
	Analytical	Numerical	Analytical	Numerical	Analytical	Numerical
12	.36240	.3333	-.12254	-.0923	.38256	.3458
24	.38386	.2960	-1.4046	-1.3900	1.4561	1.4211
36	1.0497	.9201	-2.5006	-2.5498	2.7121	2.7107
48	2.1858	2.0557	-3.1163	-3.2526	3.8065	3.8478
60	3.4922	3.4079	-3.0712	-3.2955	4.6506	4.7407
72	4.6025	4.6153	-2.3567	-2.6510	5.1708	5.3225
84	5.1859	5.3487	-1.1674	-1.4785	5.3157	5.5493
96	5.0690	5.4012	.1436	-.0821	5.0710	5.4019
108	4.3066	4.7449	1.1909	1.1694	4.4682	4.8869
120	3.1473	3.5384	1.6984	1.9413	3.5763	4.0359
132	1.9200	2.0857	1.5746	2.0193	2.4831	2.9031
144	.9222	.7584	.8944	1.3657	1.2847	1.5621
156	.3649	-.0987	-.1627	.1291	.3996	.1625
168	.3660	-.2545	-1.3641	-1.3910	1.4123	1.4141

TABLE III

$a = 17{,}000$ $x_0 = 1.0$ km $\dot{x}_0 = .01$ km/s

$e = .001$ $y_0 = 1.0$ km $\dot{y}_0 = .01$ km/s $J_2 \neq 0$

Time (hrs.)	x (km)		y (km)		R (km)	
	Analytical	Numerical	Analytical	Numerical	Analytical	Numerical
12	-5.4330	-5.7265	-10.196	-9.8802	11.553	11.419
24	-5.3963	-6.2676	-23.018	-22.842	23.642	23.686
36	1.0243	-.2816	-34.106	-34.553	34.121	34.555
48	12.203	10.842	-40.543	-41.893	42.340	43.273
60	25.264	24.313	-40.459	-42.799	47.699	49.223
72	36.598	36.636	-33.621	-36.851	49.697	51.963
84	42.819	44.512	-21.800	-25.427	48.049	51.263
96	41.988	45.725	-8.4619	-11.394	42.832	47.123
108	34.453	39.748	+2.4078	1.6073	34.537	39.781
120	22.672	27.941	7.8078	10.105	23.978	29.712
132	10.097	13.236	6.6514	11.713	12.091	17.674
144	-.0809	- .5861	-.3015	5.7899	.3122	5.8194
156	-5.6315	-9.8678	-11.119	-6.3433	12.464	11.731
168	-5.3791	-12.038	-23.316	-21.695	23.929	24.811

TABLE IV

$a = 20,000$ km $\quad x_0 = 1.0$ km $\quad \dot{x}_0 = .01$ km/s

$e = .001 \quad y_0 = 1.0$ km $\quad \dot{y}_0 = .01$ km/s $\quad J_2 \neq 0$

Time (hrs.)	x (km)		y (km)		R (km)	
	Analytical	Numerical	Analytical	Numerical	Analytical	Numerical
12	12.439	12.660	8.3138	8.4921	14.962	15.244
24	25.827	26.500	10.169	10.286	27.757	28.427
36	38.758	39.816	6.2874	5.9627	39.265	40.260
48	48.921	50.074	-2.6480	-3.6902	48.992	50.210
60	54.447	55.232	-15.092	-16.867	56.500	57.750
72	54.228	54.290	-28.825	-31.037	61.413	62.536
84	48.133	47.331	-41.266	-43.513	63.399	64.293
96	37.261	35.719	-49.970	-51.846	62.333	62.959
108	23.674	21.564	-53.150	-54.426	58.184	58.543
120	10.110	7.6823	-50.249	-50.722	51.257	51.301
132	-.8132	-3.3527	-41.913	-41.372	41.921	41.507
144	-7.1295	-9.3127	-29.965	-28.210	30.801	29.707
156	-7.9050	-9.0567	-16.673	-13.669	18.452	16.397
168	-3.2543	-2.5861	-4.4052	-.6384	5.4769	2.6637

TABLE V

$a = 20{,}000$ km $\quad x_0 = 1.0$ km $\quad \dot{x}_0 = .001$ km/s

$e = .0001 \quad y_0 = 1.0$ km $\quad \dot{y}_0 = .001$ km/s $\quad J_2 \neq 0$

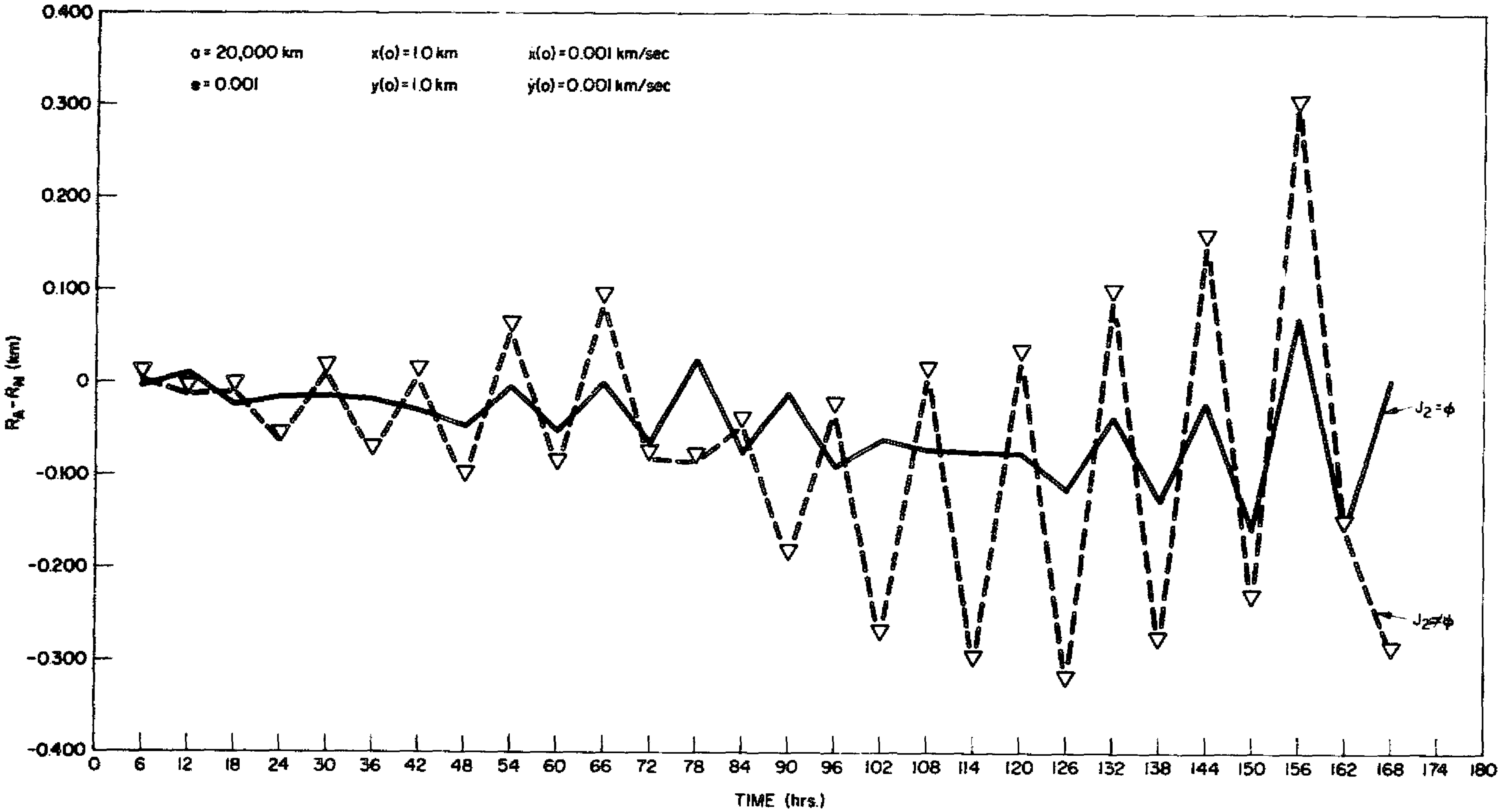
Time (hrs.)	x (km)		y (km)		R (km)	
	Analytical	Numerical	Analytical	Numerical	Analytical	Numerical
12	2.1438	2.1665	1.7308	1.7477	2.7553	2.7835
24	3.4827	3.5514	1.9163	1.9260	3.9751	4.0401
36	4.7758	4.8835	1.5276	1.4920	5.0142	5.1064
48	5.7921	5.9102	.6338	.5255	5.8266	5.9335
60	6.3448	6.4264	-.6109	-.7937	6.3742	6.4753
72	6.3228	6.3330	-1.9846	-2.2119	6.6269	6.7082
84	5.7135	5.6376	-3.2285	-3.4610	6.5626	6.6152
96	4.6260	4.4772	-4.0999	-4.2955	6.1814	6.2046
108	3.2675	3.0623	-4.4180	-4.5549	5.4950	5.4887
120	1.9108	1.6748	-4.1286	-4.1858	4.5494	4.5085
132	.8187	.5720	-3.2951	-3.2521	3.3952	3.3020
144	.1867	-.0231	-2.1009	-1.9373	2.1091	1.9374
156	.1095	.0033	-.7718	-.4845	.7795	.4845
168	.5743	.6510	.4544	.8169	.7323	1.0446

TABLE VI

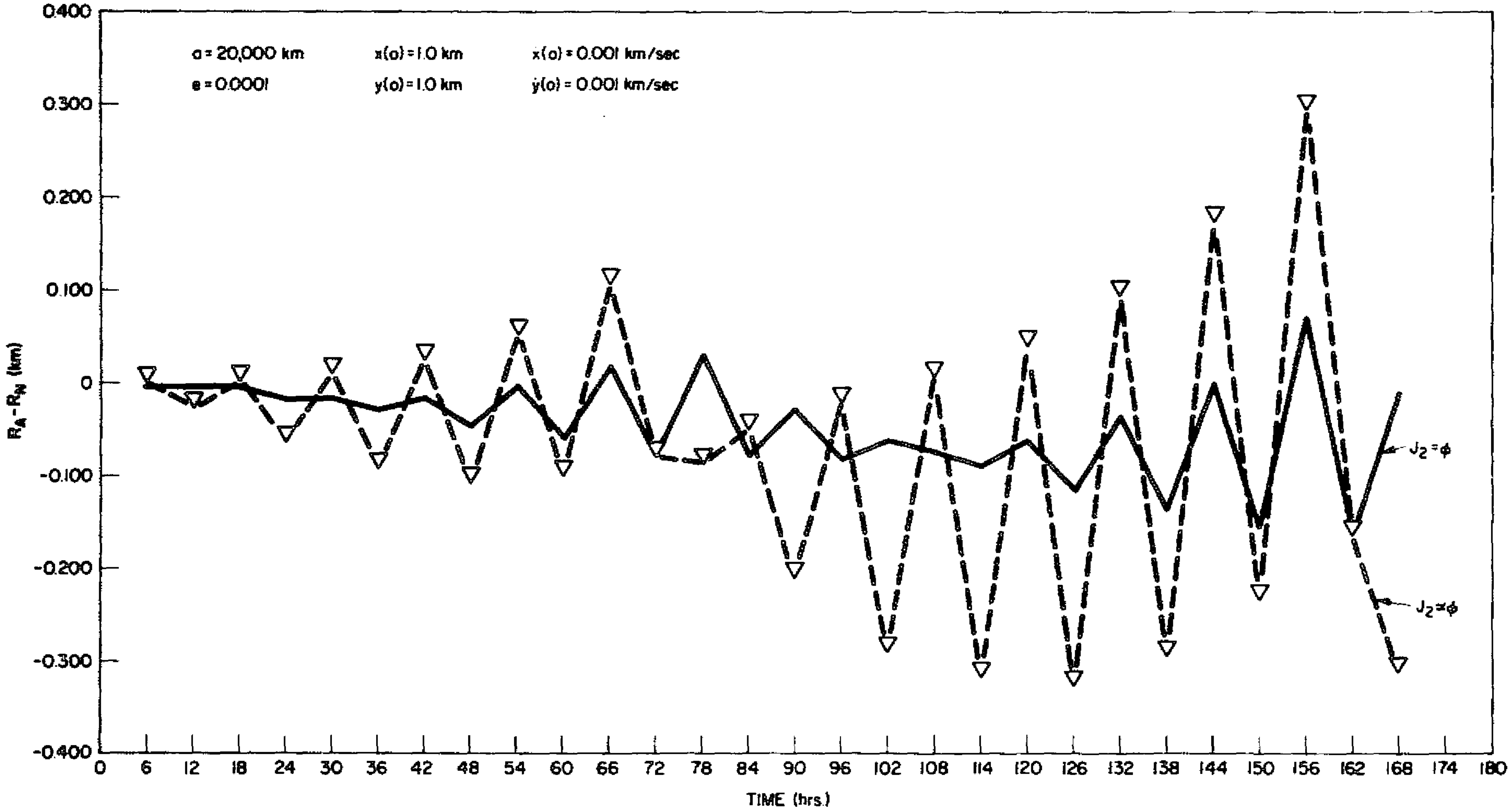
$a = 20{,}000$ km $\quad x_0 = 1.0 \quad \dot{x}_0 = .001$

$e = .0001 \quad y_0 = 1.0 \quad \dot{y}_0 = .01 \quad J_2 \neq 0$

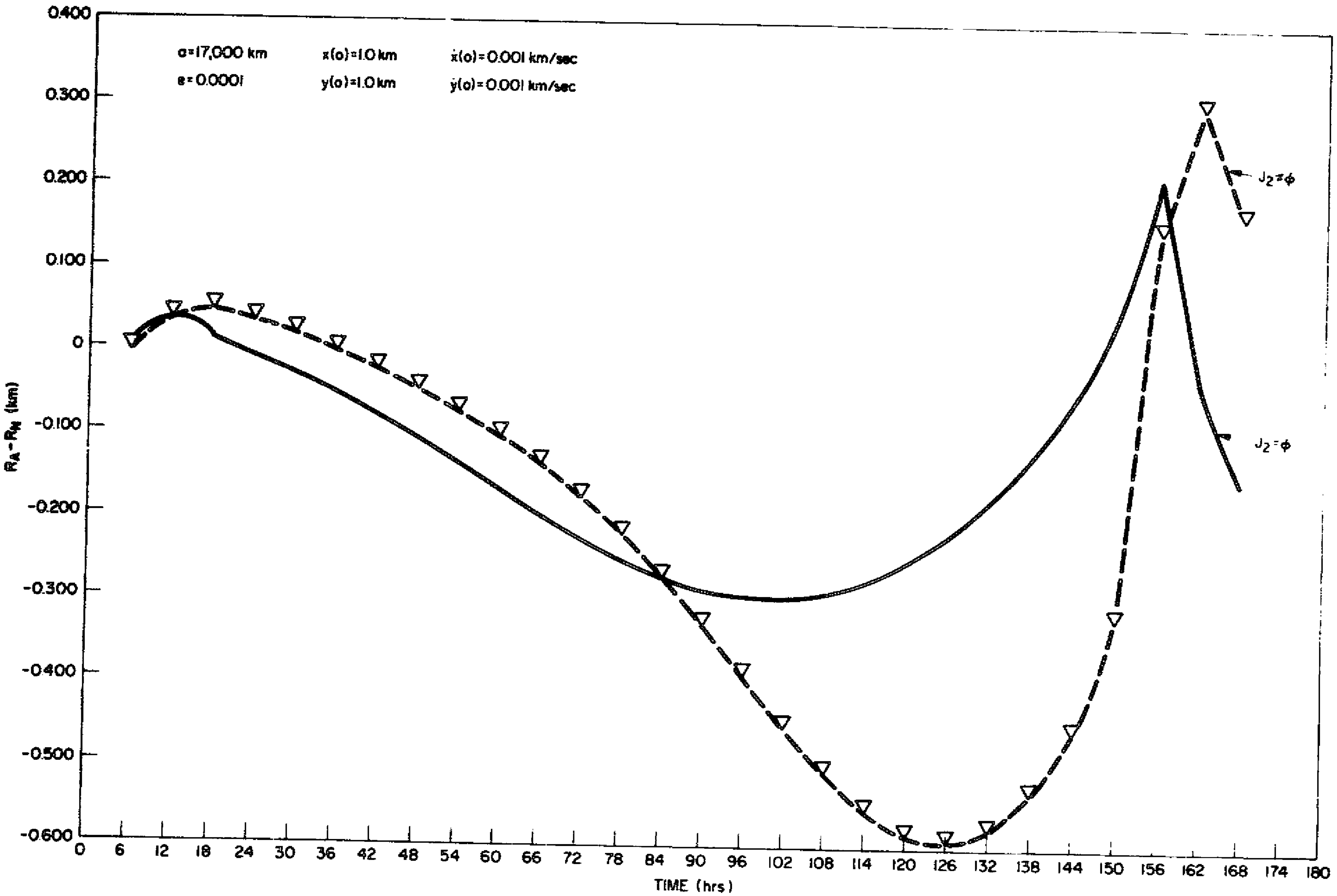
Time (hrs.)	x (km)		y (km)		R (km)	
	Analytical	Numerical	Analytical	Numerical	Analytical	Numerical
12	3.9966	4.0336	10.166	10.359	10.924	11.116
24	10.520	10.830	17.208	17.565	20.169	20.636
36	19.376	20.092	20.887	21.170	28.490	29.187
48	28.982	30.070	20.542	20.469	35.524	36.375
60	37.617	38.828	16.181	15.534	40.950	41.820
72	43.672	44.678	8.5252	7.3078	44.497	45.272
84	45.913	46.456	-1.0618	-2.6941	45.925	46.534
96	43.848	43.812	-10.748	-12.511	45.146	45.563
108	37.802	37.180	-18.614	-20.308	42.136	42.365
120	29.032	27.871	-23.127	-24.527	37.118	37.126
132	19.269	17.586	-23.462	-24.359	30.361	30.044
144	10.414	8.3780	-19.737	-19.811	22.316	21.510
156	4.0032	1.9678	-12.779	-11.708	13.392	11.872
168	1.0361	-.3454	-3.9434	-1.6383	4.0772	1.6743



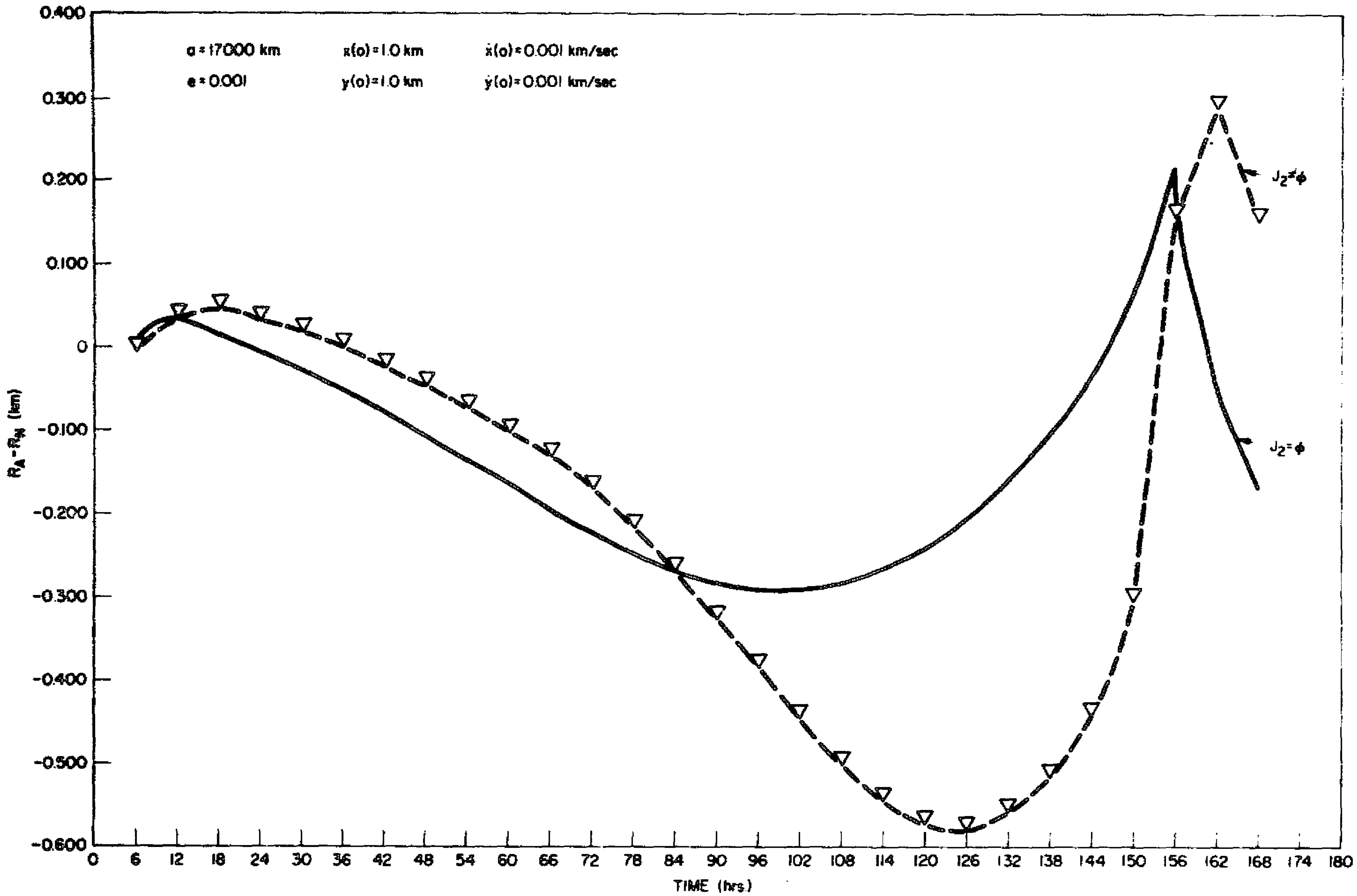
a = 20,000 km
e = 0.001
x(o) = 1.0 km
y(o) = 1.0 km
ẋ(o) = 0.001 km/sec
ẏ(o) = 0.001 km/sec
$R_A - R_N$ (km)
TIME (hrs.)
$J_2 = \phi$
$J_2 \neq \phi$



a = 20,000 km
e = 0.0001
x(o) = 1.0 km
y(o) = 1.0 km
ẋ(o) = 0.001 km/sec
ẏ(o) = 0.001 km/sec
$J_2 = \phi$
$J_2 \neq \phi$
$R_A - R_N$ (km)
TIME (hrs)



a=17,000 km
e=0.0001
x(o)=1.0 km
y(o)=1.0 km
ẋ(o)=0.001 km/sec
ẏ(o)=0.001 km/sec
$J_2 \neq \phi$
$J_2 = \phi$
$R_A - R_N$ (km)
0.400
0.300
0.200
0.100
0
-0.100
-0.200
-0.300
-0.400
-0.500
-0.600
0 6 12 18 24 30 36 42 48 54 60 66 72 78 84 90 96 102 108 114 120 126 132 138 144 150 156 162 168 174 180
TIME (hrs)



a = 17000 km
x(o) = 1.0 km
ẋ(o) = 0.001 km/sec
e = 0.001
y(o) = 1.0 km
ẏ(o) = 0.001 km/sec
$J_2 \neq \phi$
$J_2 = \phi$
$R_A - R_M$ (km)
0.400
0.300
0.200
0.100
0
-0.100
-0.200
-0.300
-0.400
-0.500
-0.600
0 6 12 18 24 30 36 42 48 54 60 66 72 78 84 90 96 102 108 114 120 126 132 138 144 150 156 162 168 174 180
TIME (hrs)

APPENDIX I

PRECEDING PAGE BLANK NOT FILMED

APPENDIX I

The functions appearing (20) - (25) are

$$\alpha_0 = \frac{D_2\, x(0) - B_1\, \dot{y}(0)}{A_1 D_2 - B_1 C_2}$$

$$\alpha_1 = \frac{B_2\, y(0) - D_1\, \dot{x}(0)}{B_2 C_1 - A_2 D_1}$$

$$\alpha_2 = \frac{A_1\, \dot{y}(0) - C_2\, x(0)}{A_1 D_2 - B_1 C_2}$$

$$\alpha_3 = \frac{C_1\, \dot{x}(0) - A_2\, y(0)}{B_2 C_1 - A_2 D_1}$$

where

$$A_1 = \frac{1}{2\bar{n}} - e\,\frac{\bar{n}}{2n}\,\frac{4\bar{n} - n}{4\bar{n}^2 - n^2}$$

$$+ \epsilon\left[\frac{3}{16}\left(\frac{R_e}{a}\right)^2 \frac{\bar{n}\sin^2\iota}{\bar{n}^2 - (w_0 + n)^2} + \frac{1}{16}\left(\frac{R_e}{a}\right)^2 (\bar{n})^{-1}\left(1 - \frac{3}{2}\sin^2\right)\right]$$

$$B_1 = 1 + e\,\frac{\bar{n}}{n}\,\frac{4\bar{n}^2 + n\bar{n} - 8n^2}{4\bar{n}^2 - n^2}$$

$$+ \epsilon\left(\frac{R_e}{a}\right)^2 \frac{\bar{n}^2}{16(w_0 + n)}\left[\frac{3(w_0 + n) - 5\bar{n}}{(w_0 + n - \bar{n})\,[2\bar{n} - (w_0 + n)]} - \frac{3(w_0 + n) + 5\bar{n}}{(w_0 + n + \bar{n})(2\bar{n} + w_0 + n)}\right]\sin^2$$

PRECEDING PAGE BLANK NOT FILMED

$$C_1 = 1 + e \frac{\bar{n}}{2n} \frac{8\bar{n}^2 - n\bar{n} - 16n^2}{(2\bar{n}+n)(2\bar{n}-n)} + \epsilon \left\{ \frac{\tilde{\omega}_z}{\bar{n}} \right.$$

$$- \frac{1}{8}\left(\frac{R_e}{a}\right)^2 \bar{n}^2 \sin^2 i \left[\frac{1}{(w_0 + n + \bar{n})^2} + \frac{1}{(w_0 + n - \bar{n})^2} \right]$$

$$- \frac{1}{16}\left(\frac{R_e}{a}\right)^2 \frac{\bar{n}^2 \sin^2 i}{w_0 + n} \left[\frac{3(w_0 + n)\bar{n} + 5\bar{n}^2}{(w_0 + n + \bar{n})^2 (2\bar{n} + w_0 + n)} \right.$$

$$\left.\left. + \frac{3(w_0 + n)\bar{n} - 5\bar{n}^2}{(w_0 + n - \bar{n})^2 [2\bar{n} - (w_0 + n)]} \right] \right\}$$

$$D_1 = 1 - e \frac{\bar{n}^2}{n} \frac{4\bar{n} - n}{4\bar{n}^2 - n^2} - \epsilon \left(\frac{R_e}{a}\right)^2 \frac{\bar{n}^2 \sin^2 i}{(w_0 + n)^2} \frac{\bar{n}^2 - 2(w_0 + n)^2}{4[\bar{n}^2 - (w_0 + n)^2]}$$

$$A_2 = 2\bar{n} + e \frac{2\bar{n}^2}{n}$$

$$- \epsilon \frac{1}{8}\left(\frac{R_e}{a}\right)^2 \frac{\bar{n}^2 \sin^2 i}{w_0 + n} \left[\frac{3(w_0 + n) + 5\bar{n}}{2\bar{n} + w_0 + n} \right.$$

$$\left. + \frac{3(w_0 + n) - 5\bar{n}}{2\bar{n} - (w_0 + n)} \right]$$

$$B_2 = e \frac{2 \bar{n}^2 (\bar{n} - n)}{4 \bar{n}^2 - n^2} + \epsilon \frac{1}{2} \left(\frac{R_e}{a}\right)^2 \frac{\bar{n}^3 \sin^2 \iota}{\bar{n}^2 - (w_0 + n)^2}$$

$$C_2 = - e \frac{\bar{n} (\bar{n} - n)}{4 \bar{n}^2 - n^2} - \epsilon \left[\frac{\tilde{\omega}_z}{\bar{n}} + \frac{3}{8} \left(\frac{R_e}{a}\right)^2 \frac{\bar{n}^2 \sin^2 \iota}{\bar{n}^2 - (w_0 + n)^2}\right]$$

$$D_2 = - 2 \bar{n} - e \frac{2 \bar{n}^2}{n} + \epsilon \left\{- 2 \tilde{\omega}_z\right.$$

$$+ \frac{1}{2} \left(\frac{R_e}{a}\right)^2 \frac{\bar{n}^3 \sin^2 \iota}{\bar{n}^2 - (w_0 + n)^2}$$

$$+ \frac{1}{8} \left(\frac{R_e}{a}\right)^2 \frac{\bar{n}^3 \sin^2 \iota}{w_0 + n} \left[\frac{3 (w_0 + n) + 5 \bar{n}}{(w_0 + n + \bar{n}) (2 \bar{n} + w_0 + n)}\right.$$

$$\left.\left. - \frac{3 (w_0 + n) - 5 \bar{n}}{(w_0 + n - \bar{n}) (2 \bar{n} - (w_0 + n))}\right]\right\}$$

The coefficients of Equation (22) are

$$\beta^{01}_{1x} = 2 \frac{\bar{n}^2}{n} \frac{\bar{n} - n}{4 \bar{n}^2 - n^2} \alpha_3$$

$$\beta^{01}_{2x} = - \frac{\bar{n}}{2 n} \frac{4 \bar{n} - n}{4 \bar{n}^2 - n^2} \alpha_0$$

$$\beta_{3x}^{01} = + \frac{\bar{n}}{2\,n}\;\frac{\bar{n} + 8\,n}{2\,\bar{n} + n}\;\alpha_1$$

$$\beta_{4x}^{01} = \frac{\bar{n}}{2\,n}\;\frac{3\,\bar{n} - 8\,n}{2\,\bar{n} - n}\;\alpha_1$$

$$\beta_{5x}^{01} = \frac{\bar{n}}{2\,n}\;\frac{\bar{n} + 8\,n}{2\,\bar{n} + n}\;\alpha_2$$

$$\beta_{6x}^{01} = \frac{\bar{n}}{2\,n}\;\frac{3\,\bar{n} - 8\,n}{2\,\bar{n} - n}\;\alpha_2$$

The coefficients of Equation (23) are

$$\beta_{1y}^{01} = \frac{\bar{n}}{n}\;\frac{n - \bar{n}}{4\,\bar{n}^2 - n^2}\;\alpha_0$$

$$\beta_{2y}^{01} = \left(\frac{\bar{n}}{n}\right)^2\;\frac{n\,(n - 4\,\bar{n})}{4\,\bar{n} - n^2}\;\alpha_3$$

$$\beta_{3y}^{10} = - \frac{\bar{n}^2 + 8\,n\,\bar{n}}{2\,n\,(2\,\bar{n} + n)}\;\alpha_2$$

$$\beta_{4y}^{10} = - \frac{3\,\bar{n}^2 - 8\,n\,\bar{n}}{2\,n\,(2\,\bar{n} - n)}\;\alpha_2$$

$$\beta_{5y}^{10} = \frac{\bar{n}^2 + 8\,n\,\bar{n}}{2\,n\,(2\,\bar{n} + n)}\;\alpha_1$$

$$\beta_{6y}^{10} = \frac{3\,\bar{n}^2 - 8\,n\,\bar{n}}{2\,n\,(2\,\bar{n} - n)}\;\alpha_1$$

The coefficients of Equation (24) are

$$\beta_{1x}^{10} = \frac{1}{4}\left(\frac{R_e}{a}\right)^2 \frac{\bar{n}^3 \sin^2 \iota}{(w_0 + n)\ [\bar{n}^2 - (w_0 + u)^2]}\ \alpha_3$$

$$\beta_{2x}^{10} = \frac{3}{16}\left(\frac{R_e}{a}\right)^2 \frac{\bar{n} \sin^2 \iota}{\bar{n}^2 - (w_0 + n)^2}\ \alpha_0$$

$$\beta_{3x}^{10} = -\frac{1}{16}\left(\frac{R_e}{a}\right)^2 \frac{\bar{n}^2 \sin^2 \iota}{(w_0 + n)\ [2\bar{n} + (w_0 + n)]}\ \frac{3(w_0 + n) + 5\bar{n}}{w_0 + n + \bar{n}}\ \alpha_1$$

$$\beta_{4x}^{10} = -\frac{1}{16}\left(\frac{R_e}{a}\right)^2 \frac{\bar{n}^2 \sin^2 \iota}{(w_0 + n)\ [2\bar{n} - (w_0 + n)]}\ \frac{3(w_0 + n) - 5\bar{n}}{w_0 + n - \bar{n}}\ \alpha_1$$

$$\beta_{5x}^{10} = -\frac{1}{16}\left(\frac{R_e}{a}\right)^2 \frac{\bar{n}^2 \sin^2 \iota}{(w_0 + n)\ [2\bar{n} + (w_0 + n)]}\ \frac{3(\omega_0 + n) + 5\bar{n}}{w_0 + n + \bar{n}}\ \alpha_2$$

$$\beta_{6x}^{10} = \frac{1}{16}\left(\frac{R_e}{a}\right)^2 \frac{\bar{n}^2 \sin^2 \iota}{(w_0 + n)\ [2\bar{n} - (w_0 + n)]}\ \frac{3(w_0 + n) - 5\bar{n}}{w_0 + n - \bar{n}}\ \alpha_2$$

$$\beta_{7x}^{10} = \frac{1}{16\bar{n}}\left(\frac{R_e}{a}\right)^2 \left(1 - \frac{3}{2}\sin^2 \iota\right)\ \alpha_0$$

The coefficients of Equation (25) are

$$\beta_{1y}^{10} = -\frac{\tilde{\omega}_z}{\bar{n}}\,\alpha_2$$

$$\beta_{2y}^{10} = \frac{\tilde{\omega}_z}{\bar{n}}\,\alpha_1$$

$$\beta_{3y}^{10} = -\frac{3}{16}\left(\frac{R_e}{a}\right)^2 \frac{\bar{n}^2 \sin^2 \iota}{(w_0 + n)\,[\bar{n}^2 - (w_0 + n)^2]}\,\alpha_0$$

$$\beta_{4y}^{10} = \frac{1}{4}\left(\frac{R_e}{a}\right)^2 \frac{\bar{n}^2 \sin^2 \iota}{(w_0 + n)^2}\,\frac{2\,(w_0 + n)^2 - \bar{n}^2}{\bar{n}^2 - (w_0 + n)^2}\,\alpha_3$$

$$\beta_{5y}^{10} = \frac{1}{8}\left(\frac{R_e}{a}\right)^2 \frac{\bar{n}^2 \sin^2 \iota}{(w_0 + n + \bar{n})^2}\left\{1 + \frac{3\,\bar{n}\,(w_0 + n) + 5\,\bar{n}^2}{2\,(w_0 + n)\,(2\,\bar{n} + w_0 + n)}\right\}\alpha_2$$

$$\beta_{6y}^{10} = -\frac{1}{8}\left(\frac{R_e}{a}\right)^2 \frac{\bar{n}^2 \sin^2 \iota}{(w_0 + n - \bar{n})^2}\left\{1 + \frac{3\,\bar{n}\,(w_0 + n) - 5\,\bar{n}^2}{2\,(w_0 + n)\,(2\,\bar{n} - w_0 - n)}\right\}\alpha_2$$

$$\beta_{7y}^{10} = -\frac{1}{8}\left(\frac{R_e}{a}\right)^2 \frac{\bar{n}^2 \sin^2 \iota}{(w_0 + n + \bar{n})^2}\left\{1 + \frac{3\,\bar{n}\,(w_0 + n) + 5\,\bar{n}^2}{2\,(w_0 + n)\,(2\,\bar{n} + w_0 + n)}\right\}\alpha_1$$

$$\beta_{8y}^{10} = -\frac{1}{8}\left(\frac{R_e}{a}\right)^2 \frac{\bar{n}^2 \sin^2 \iota}{(w_0 + n - \bar{n})^2}\left\{1 + \frac{3\,\bar{n}\,(w_0 + n) - 5\,\bar{n}^2}{2\,(w_0 + n)\,(2\,\bar{n} - w_0 - n)}\right\}\alpha_1$$

$$\beta_{9y}^{10} = -2\,\frac{\tilde{\omega}_z}{\bar{n}}\,\alpha_0$$

APPENDIX II

APPENDIX II

NUMERICAL PROGRAM

General Overview

The numerical program is comprised of a MAIN routine and six (6) subroutines. The MAIN routine reads in the data (via 2 namelists) and writes out the initial conditions. Execution starts by integrating the equations of motion for the desired time period. Values for the orbital elements are written for each time point. Upon completion of the integration the values for position and velocity are plotted vs. time on the SD 4060.

PRECEDING PAGE BLANK NOT FILMED

MAIN Routine Description

Purpose: To store initial values read in of the variables to be integrated. The variables not read in are calculated from input variables. The following variables are integrated: $\dot{a}, \dot{e}, \dot{\ell}, \dot{m}, \dot{u}, \dot{\Omega}, \dot{x}, \dot{y}, \dot{z}, \ddot{x}, \ddot{y}, \ddot{z}$.

ℓ and m are obtained from e and w by:

$\ell = e \cos w$

$m = e \sin w$

u is defined as $u = v + w$.

The integrator (FNOL2) is then called. All values are calculated for each time point until the specified final time is reached.

Subroutine DERIV Description

Purpose: To compute values of the differentials to be used by the integrator (FNOL2)

Method: The systems of differential equations $\{\dot{a}, \dot{\iota}, \dot{\Omega}, \dot{\ell}, \dot{m}, \dot{u}\}$ and $\{\dot{x}, \dot{y}, \dot{z}, \ddot{x}, \ddot{y}, \ddot{z}\}$ are calculated from Equations (27), (9) and (10). The intermediate variables used in the integration of these equations are obtained from expressions for

p	$\ddot{\iota}$
n	$\dot{\omega}_x$
$\dot{x}_0$	$\dot{\omega}_z$
$\dot{p}$	
ω_z	
ω_x	

These expressions are to be found in the listing of the numerical program in Appendix 4.

Subroutine Description

I. MAIN

Purpose: To read in control parameters and initial conditions.

To write out control parameters and initial conditions.

Calls: DATE
FNOL2
PLOT

Method: All input through use of namelists

COMMON blocks used: INCOND
PLOTS
CONST
FLAG

Variables not in COMMON:

FORTRAN Name	Format	Description
A	R*8	semi-major axis
E	R*8	eccentricity
I	R*8	inclination
J	I*4	no. of time points
R (500)	R*8	position
T (500)	R*8	time
V	R*8	true anomaly
W	R*8	argument of perigee
X	R*8	X-coord.
Y	R*8	Y-coord.
Z	R*8	Z-coord.
DR (500)	R*8	velocity
NO	I*4	'NO'
PL	I*4	control parameter to PLOT or not to
PR	I*4	control parameter to print intermediate computed values
TF	R*8	final time
TI	R*8	initial time
XO	R*8	distance between origins of 2 coordinate systems (XO on fig.)

FORTRAN name	Format	Description
FMU	R*8	μ = 3.986032 D5
J20	R*8	J_{20} = 1.0823D-3
RSQ	R*8	r^2 = 4.068098 D7
YES	I*4	'YES'
DELM (20)	R*8	derivative array
ELEM (20)	R*8	integrated array
STEP	R*8	integration interval
XDOT	R*8	X-component of velocity
YDOT	R*8	Y-component of velocity
ZDOT	R*8	Z-component of velocity
EPOCH	R*8	epoch date in form: YYMMDD.D
KPLOT	L*4	plot or not
OMEGA	R*8	longitude of ascending node
KOMEGA	I*4	parameter to set $\omega_x = 0$.
KPRINT	I*4	parameter to print intermediate values
KUTMOL	I*4	type of integration scheme to use

II. DERIV

Purpose: To define variables to be integrated

Called by: FNOL2

Method: Variables to be integrated in equations in terms of predefined variables.

COMMON blocks used: CONST
FLAG
ALFA
ANS

Variables not in COMMON:

FORTRAN name	Format	Description
A	R*8	semi-major axis
C	R*8	variable used to compute $\dot{v}$
E	R*8	eccentricity
I	R*8	inclination
N	R*8	mean motion
P	R*8	semi-latus rectum
T	R*8	time
U	R*8	v + w
V	R*8	true anomaly
X	R*8	X-coord.
Y	R*8	Y-coord.
A1	R*8	A_1, intermediate parameter
A2	R*8	A_2, intermediate parameter
B1	R*8	B_1, intermediate parameter
B2	R*8	B_2, intermediate parameter
C1	R*8	C_1, intermediate parameter
C2	R*8	C_2, intermediate parameter
DA	R*8	$\dot{a}$, differential of semi-major axis
DE	R*8	$\dot{e}$, differential of eccentricity
DI	R*8	$\dot{i}$, differential of inclination
DP	R*8	$\dot{p}$, differential of semi-latus rectum
DV	R*8	$\dot{V}$, differential of true anomaly
DX	R*8	$\dot{X}$, x-comp of velocity
DY	R*8	$\dot{Y}$, y-comp of velocity
D1	R*8	D_1, intermediate parameter
D2	R*8	D_2, intermediate parameter

FORTRAN name	Format	Description
E2 (200)	R*8	dummy array of integrated values
OM	R*8	Ω, longitude of ascending node
WO	R*8	w_0, argument of perigee
XO	R*5	X_0, distance between origins of
DEL (200)	R*8	dummy array of differentials
DOM	R*8	$\dot{\Omega}$, differential of long. of ascending node
DWO	R*8	w_0, differential of arg. of perigee
EPS	R*8	ϵ
FJ2	R*8	J_{20}
FMU	R*8	μ = 3.986032 D5
NEG	I*4	no. of parameters to integrate
NMN	R*8	$\bar{n} - n$
NPN	R*8	$\bar{n} + n$
NPO	R*8	$n + w_0$
NSS	R*8	$n^2 \sin^2 \iota$
OZT	R*8	$\tilde{\omega}_z$
RDA	R*8	r^2/a^2
RSQ	R*8	r^2, square of earth's radius
TXO	R*8	intermediate parameter
TYO	R*8	intermediate parameter
X00	R*8	X_{00}, intermediate parameter
X01	R*8	X_{01}, intermediate parameter
X10	R*8	X_{10}, intermediate parameter
Y00	R*8	Y_{00}, intermediate parameter
Y01	R*8	Y_{01}, intermediate parameter
Y10	R*8	Y_{10}, intermediate parameter
ALF0	R*8	α_0, intermediate parameter
ALF1	R*8	α_1, intermediate parameter
ALF2	R*8	α_2, intermediate parameter
ALF3	R*8	α_3, intermediate parameter
COSI	R*8	cos (i) intermediate parameter
COSU	R*8	cos (u) intermediate parameter
COSV	R*8	cos (v) intermediate parameter
COTI	R*8	cotan (i) intermediate parameter
DELM (200)	R*8	array of derivatives
DOCI	R*8	$\dot{\Omega}$. cos i, intermediate parameter
DXOO	R*8	$\dot{X}_{00}$
DX01	R*8	$\dot{X}_{01}$
DX10	R*8	$\dot{X}_{10}$
DY00	R*8	$\dot{Y}_{00}$
DY01	R*8	$\dot{Y}_{01}$

FORTRAN Name	Format	Description
DY10	R*8	$\dot{Y}_{10}$
ELEM (200)	R*8	array of integrated variables
LAM1	R*8	λ_1^{01}
LAM2	R*8	λ_2^{01}
LAM3	R*8	λ_3^{01}
LAM4	R*8	λ_4^{01}
LAM5	R*8	λ_5^{01}
LAM6	R*8	λ_6^{01}
LAM7	R*8	λ_3^{10}
LAM8	R*8	λ_4^{10}
LAM9	R*8	λ_5^{10}
LAM10	R*8	λ_6^{10}
LAM11	R*8	λ_7^{10}
LAM12	R*8	λ_8^{10}
NBAR	R*8	$\bar{n}$
SINI	R*8	sin (i)
SINU	R*8	sin (u)
SINV	R*8	sin (v)
TANU	R*8	tan (u)
TDXO	R*8	$\dot{X}$ (0), initial value of X-component of velocity
TDYO	R*8	$\dot{Y}$ (0), initial value of Y-component of velocity
TNMN	R*8	$2\bar{n} - n$
TNPN	R*8	$2\bar{n} + n$
ZERO (4)	R*8	array of initial values of position and velocity
COS2I	R*8	cos (2i)
COS2U	R*8	cos (2u)
SIN2I	R*8	sin (2i)
SIN2U	R*8	sin (2u)

APPENDIX III

APPENDIX III

ANALYTIC PROGRAM

General Overview

The analytic program is comprised of a MAIN routine and four (4) subroutines. The MAIN routine reads in the data (via 2 namelists) and writes out the initial conditions. Execution starts by integrating the equations of motion for the desired time period. Values for the orbital elements are written for each time point.

MAIN Routine Description

Purpose: To store initial values read in of the variables to be integrated. The variables not read in are calculated from those that are. The following are the variables that are integrated:

$\dot{a}$, $\dot{e}$, $\dot{i}$, $\dot{\Omega}$, $\dot{u}$, $\dot{w}_0$,

u is obtained from:

$u = v + w_0$

The integrator is then called. All values are calculated for each time point until the specified final time is reached.

Subroutine DERIV Description

Purpose: To compute values of the differentials to be used by the integrator (FNOL2).

Method: $\dot{a}$, $\dot{e}$, $\dot{i}$, $\dot{\Omega}$, $\dot{u}$, $\dot{w}_0$ are calculated by equations (30), (31), respectively. The intermediate variables used to determine position and velocity are obtained from:

x_0
ϵ
$\bar{n}$
$\tilde{\omega}_z$
$A_1, B_1, C_1, D_1;$
$A_2, B_2, C_2, D_2;$
$\alpha_0, \alpha_1, \alpha_2, \alpha_3;$
$\lambda_1, \lambda_2, \lambda_3, \ldots, \lambda_{12}$
$X_{00}, X_{01}, X_{10};$
$Y_{00}, Y_{01}, Y_{10};$
$\dot{X}_{00}, \dot{X}_{01}, \dot{X}_{10}$
$\dot{Y}_{00}, \dot{Y}_{01}, \dot{Y}_{10}$

These expressions are to be found in the listing of the analytical program in Appendix V.

Subroutine Description

I. MAIN

A. Purpose: To read in control parameters and initial conditions.

To write out control parameters and initial conditions.

B. Calls: FNOL2

C. Method: All input through use of namelists:

1. INIT:

Variable	Variable Name	Format	Description
$X(0)$, $\dot{X}(0)$, $Y(0)$, $\dot{Y}(0)$	ZERO (4)	R*8	initial value of pos & vel.
a	A	R*8	semi-major axis
e	E	R*8	eccentricity
i	I	R*8	inclination
v	V	R*8	true anomaly
Ω	OM	R*8	longitude of node
w_0	WO	R*8	argument of perigee
J_{20}	FJ2	R*8	oblateness constant

2. INTEG

Variable	Variable Name	Format	Description
t_f	TF	R*8	final time
Δt	TINCR	R*8	integration time step
t_i	TI	R*8	initial time
	INTP	I*4	print interval
	KUTMOL	I*4	integration type
	PRINT	L*1	flag for intermediate values
	NEQ	I*4	no. of variables to integrate

B. Common blocks used:

1. ELEMENT

Variable	FORTRAN Name	Format	Description
a	A	R*8	semi-major axis
e	E	R*8	eccentricity
i	I	R*8	inclination
Ω	OM	R*8	longitude of node
v	V	R*8	true anomaly
u	U	R*8	
w_0	WO	R*8	argument of perigee neglecting periodic behavior

B. Common blocks used:

1. ELEMENT
   A, E, I, OM, V, U, WO
2. INTEG
   TF, TINCR, TI, INTP, KUTMOL
3. CONST
   FJ2, FMU, RSQ, ZERO
4. FLAG
   PRINT, J, NEQ
5. ANS
   X, Y, XO, DX, DY

II. DERIV

A. Purpose: To define variables to be integrated (differentials)

B. Called by: FNOL2

C. Method: Differentials defined in terms of variables previously defined

D. COMMON blocks used: CONST: FJ2, FMU, RSQ, ZERO
FLAG: PRINT, J, NEQ
ANS: X, Y, XO, DX, DY
ALFA: ALFO, ALF1, ALF2, ALF3

APPENDIX IV

PRECEDING PAGE BLANK NOT FILMED

APPENDIX IV

```
C
C
C PURPOSE    TO SOLVE THE EQUATIONS OF MOTION OF TWO SATELLITES NUMERICALLY
C            I.E. COMPUTE VALUES OF POSITION( X,Y,Z,R)  VELOCITY( X',Y',Z',R' ),
C            SEMI-MAJOR AXIS( A ), ECCENTRICITY( E ), INCLINATION( I ),
C            LONGITUDE OF NODE, ARGUMENT OF PERIGEE, U,V AS A FUNCTION OF
C            TIME.
C
C PROGRAM    WRITTEN FOR R. BARBIERI BY JOEL MASHBAUM  1-1-71
C
C INPUT    BY NAMELIST
C          1. NAMELIST INTEG
C            A. STEP = INTEGRATION STEP SIZE, TIME IN SEC.  ( 600. )
C            B. TI   = INITIAL TIME, SECS  ( 0.0 )
C            C. TF   = FINAL TIME, SECS  ( 86,400 SEC = 1 DAY )
C            D. INTP = PRINT FREQUENCY, INTEGRATIONS  ( 1 = EACH STEP )
C            E. KUTMOL = METHOD OF INTEGRATION  ( 2 = RUNGE-KUTTA & ADAMS-MOULTON
C            F. EPOCH  = DATE OF INITIAL CONDITIONS  ( 720101.0 )
C            G. J20  = OBLATENESS CONSTANT  ( 0.0010823 )
C          2. NAMELIST INIT
C            A. X     = X-COORDINATE OF POSITION  ( 1.0 )
C            B. Y     = Y-COORDINATE OF POSITION  ( 1.0 )
C            C. Z     = Z-COORDINATE OF POSITION  ( 1.0 )
C            XDOT     = X-COORDINATE OF VELOCITY ( 0.1 )
C            YDOT     = Y-COORDINATE OF VELOCITY ( 0.1 )
C            ZDOT     = Z-COORDINATE OF VELOCITY ( 0.1 )
C            G. A     = SEMI-MAJOR AXIS ( 1500. KM )
C            H. E     = ECCENTRICITY    ( 0.01 )
C            I. I     = INCLINATION(DEG)    ( 45. )
C            J. OMEGA = LONGITUDE OF NODE, DEG  ( 45. )
C            K. W = ARGUMENT OF PERIGEE, DEG  ( 0.0 )
C            L. V = TRUE ANOMALY, DEG  ( 0.0 )
C            M. KOMEGA= FLAG TO SET WY & WY' = 0  ( 1 )
C            N. KPRINT= FLAG TO PRINT VARIOUS INTERMEDIATE VALUES  ( 0 )
C            O. KPLOT = FLAG TO DRAW 4060 GRAPH  ( .TRUE. )
C
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8  L,M
      REAL*8 I,J20
      LOGICAL KPLOT
      INTEGER PR,PL,YES,NO
      DIMENSION ELEM(20),DELM(20)
      DIMENSION R(5000),DR(5000), T(5000)
      DIMENSION TYPE(5,3)
      COMMON /INCOND/ A,E,I,OMEGA,W,U,X,Y,Z,XDOT,YDOT,ZDOT
      COMMON /PLOTS/ T,R,DR,J
      COMMON /CONST/ TF,STEP,TI,INTP,KUTMOL,EPOCH,FMU,J20,RSQ
      COMMON /FLAG/ XO, KOMEGA,KPRINT,KPLOT
      DATA ELEM,DELM / 40*0.D0 /
      DATA YES,NO / 'YES ','NO  ' /,PL,PR/ 'NO  ','NO  ' /
      DATA TYPE /'RUNGE-KU','TTA THRO','UGHOUT  ',2*'        ',
     1 'RUNGE-KU','TTA AND ','ADAMS-MO','ULTON   ','        ',
     2 'RUNGE-KU','TTA WITH',' ERROR C','ALCULATI','ON      '/
      NAMELIST /INTEG/ STEP,TI,TF,INTP,KUTMOL,EPOCH,J20
      NAMELIST /INIT/ X,Y,Z,XDOT,YDOT,ZDOT,A,E,I,OMEGA,W,V,KOMEGA,KPRINT
     , ,KPLOT
      RADIAN(X) = X / 57.2957795130823
      READ(5,INTEG)
      READ(5,INIT)
      I = RADIAN(I)
      W = RADIAN(W)
```

PRECEDING PAGE BLANK NOT FILMED

```
      V = RADIAN(V)
      OMEGA = RADIAN(OMEGA)
      L = E * DCOS(W)
      M = E * DSIN(W)
      U=W+V
      ELEM( 1) = X
      ELEM( 2) = Y
      ELEM( 3) = Z
      ELEM( 4) = A
      ELEM( 5) = L
      ELEM( 6) = I
      ELEM( 7) = OMEGA
      ELEM( 8) = M
      ELEM( 9) = U
      ELEM(10)=XDOT
      ELEM(11)=YDOT
      ELEM(12)=ZDOT
      LL = 0
      NE = 0
      WRITE(6,1)
    1 FORMAT( '1' )
      WRITE(6,2)
    2 FORMAT( 30X,10(1H*),10X,'INITIAL CONDITIONS',10X,10(1H*) )
      CALL DATE( EPOCH )
      WRITE(6,3) X,Y,Z
    3 FORMAT( '-POSITION VECTOR   ',10X,'X = ',D15.5,10X,'Y = ',D15.5,
     U 10X,'Z = ',D15.5 )
      WRITE(6,4) XDOT, YDOT, ZDOT
    4 FORMAT( '0VELOCITY VECTOR   ',9X,'X'' = ',D15.5,9X,'Y'' = ',D15.5,
     U 10X,'Z'' = ',D15.5 )
      WRITE(6,5) A,E,I,OMEGA,W,V,J20,KOMEGA
    5 FORMAT( /39X,'SEMI-MAJOR AXIS (A) = ',D20.10/42X,'ECCENTRICITY (E)
     1 = ',D20.10/42X,'INCLINATION (FI) = ',D20.10/23X,'LONGITUDE OF',
     2 ' ASCENDING NODE (OMEGA) = ',D20.10/35X,'ARGUMENT OF PERIGEE (W)
     3= ',D20.10/42X,'TRUE ANOMALY (V) = ',D20.10/42X,'OBLATENESS (J20)
     4= ',D20.10/52X,'OMEGAX = ',I2 )
      IF( KPLOT )  PL = YES
      IF( KPRINT.NE.0 )  PR = YES
      WRITE(6,6) PL,PR
    6 FORMAT( ////45X,'PLOT = ',A4,5X,'PRINT = ',A4 )
      WRITE(6,7) (TYPE(J,KUTMOL),J=1,5),STEP,INTP
    7 FORMAT( ////48X,'INTEGRATION PARAMETERS'/48X,22(1H-)//45X,
     1 'METHOD    ',5A8/45X,'STEPSIZE    ',D20.10/45X,'PRINT FREQUENCY
     2 ,I8 )
      WRITE (6,1)
      CALL FNOL2( KUTMOL,12,STEP,LL,INTP,NE,TI,ELEM,DELM)
      IF( KPLOT )  CALL PLOT
      WRITE(6,9)
    9 FORMAT('-END OF RUN')
      RETURN
      END
```

```
      SUBROUTINE DATE( EPOCH )
      IMPLICIT REAL*8 (A-H,O-Z)
      INTEGER*4 YR,DA,HR,SEC,YEAR
      DIMENSION MONTH(12)
      DATA MONTH/ ' JAN', ' FEB', ' MAR', ' APR', ' MAY', 'JUNE',
     U 'JULY', ' AUG', 'SEPT', ' OCT', ' NOV' , ' DEC' /
      YR = EPOCH / 1.D4
      YEAR = 1900 + YR
      J1 = EPOCH - YR * 1.D4
      MO = J1 / 1.D2
      DA = J1 - MO * 1.D2
      H1 = EPOCH - YR * 1.D4 - MO * 1.D2 - DA
      HR = H1 * 1.D2
      MIN = H1 * 1.D4 - HR * 1.D2
      SEC = H1 * 1.D6 - MIN * 1.D2 - HR * 1.D4
      WRITE(6,5) MONTH(MO),DA,YEAR,HR,MIN,SEC
    5 FORMAT( ////44X,'EPOCH = ',A4,I3,',',I4,5X,I2,' HRS  ',I2,' MINS ',
     $ ,I2,' SECS' )
      RETURN
      END
```

```
      SUBROUTINE DERIV(T,ELEM,DELM)
      IMPLICIT REAL*8(A-H,O-Z)
      REAL*8  L,M
      LOGICAL $SIN,$COS
C
C   VERSION OF 12-23-70
C   SUBROUTINE DERIV COMPUTES THE DERIVATIVES OF THE ELEMENTS ARRAY.
C
C   INPUT
C       T        - TIME
C       ELEM     - ARRAY OF ELEMENTS
C
C   OUTPUT
C       DELM     - ARRAY OF DERIVATIVES OF ELEMENTS
C
C                     ELEM                 DELM
C                     X                    X'               1
C                     Y                    Y'               2
C                     Z                    Z'               3
C                     AO                   AO'              4
C                     LO                   LO'              5
C                     IO                   IO'              6
C                     OMEGA                OMEGA'           7
C                     MO                   MO'              8
C                     UO                   UO'              9
C                     X'                   X''             10
C                     Y'                   Y''             11
C                     Z'                   Z''             12
C
      DIMENSION ELEM(1),DELM(1),ELE(20),DEL(20)
      COMMON /FLAG/ XO, KOMEGA,KPRINT,KPLOT
      COMMON /CONST/ TF,STEP,TI,INTP,KUTMOL,EPOCH,FMU,FJ2,RSQ
      EQUIVALENCE (ELE (1),X),(ELE (2),Y),(ELE (3),Z),
     1(ELE (4),A),(ELE (5),L),(ELE (6),FI),(ELE (7),OM),(ELE (8),M),
     2(ELE( 9),U),(DEL(1),DX,ELE(10)),(DEL(2),DY),(DEL(3),DZ),
     3(DEL (4),DA),(DEL (5),DL),(DEL (6),DI),(DEL (7),DOM),(DEL (8),DM),
     4(DEL( 9),DU),(DEL(10),DDX),(DEL(11),DDY),(DEL(12),DDZ)
C    RESTORE INPUT ARRAYS FOR EQUIVALENCE STATEMENT
C     DO 20 I=1,10
C     ELE(I)=ELEM(I)
C  20 DEL(I)=DELM(I)
      NAMELIST /TRIG/ COSV,SINV,COSU,SINU,COSI,SINI,COS2U,SIN2U,SIN2I,
     U COS2I,TANU,COTI
      NAMELIST /NAM1/ A2,A3,B2,B3,C2,C3
      NAMELIST /NAM2/ V,U,FI,P,XO,FN
      NAMELIST /NAM3/ DA,DL,DP,C,   DXO,DI,DOM,DM,DU,DDI
      NAMELIST /NAM4/ WX,WY,WZ,DWX,DWY,DWZ
      NAMELIST /NAM5/ DX,DY,DZ,DDX,DDY,DDZ
      DO 20 I=1,12
   20 ELE(I)=ELEM(I)
      IF( KPRINT.EQ.0 )  GOTO 500
      WRITE(6,2001)
 2001 FORMAT('0 ENTER' )
                        WRITE(6,2000) (ELEM(I),DELM(I),I=1,13)
 2000 FORMAT( 22X,'ELEMENTS',9X,'DERIVATIVES'//(10X,2D20.8))
  500 CONTINUE
      E = DSQRT( L*L + M*M )
      IF( DABS(M).LE.1.D-30 .AND. DABS(L).LE.1.D-30 )   GOTO 50
      W = DATAN2( M,L )
      GOTO 60
   50 W = 0.D0
   60 V = U - W
C
C   COMPUTE FREQUENTLY USED TRIGONOMETRIC FUNCTIONS
C
```

```
      COSV=DCOS(V)
      SINV=DSIN(V)
      COSU=DCOS(U)
      COS2U=DCOS(2.D0*U)
      SINU=DSIN(U)
      SIN2U=DSIN(2.D0*U)
      SINI=DSIN(FI)
      SIN2I=DSIN(2.D0*FI)
      COSI=DCOS(FI)
      COS2I=DCOS(2.D0*FI)
      TANU=DTAN(U)
      COTI=DCOTAN(FI)
      IF(KPRINT.EQ.1)  WRITE(6,TRIG)
      P=A*(1.D0-E*E)
      FN=DSQRT(FMU/A**3)
      DENOM = 1.D0 + L * COSU + M * SINU
      X0 = A *( 1.D0 - L*L - M*M ) / DENOM
      IF(KPRINT.EQ.1)    WRITE(6,NAM2)
      DA=-3.D0*FMU*FJ2*RSQ/(FN*DSQRT(1.D0-E*E))*(( L* SINU - M * COSU )
     *    *( 1.D0 - 3.D0 * SINI *
     1SINI*SINU*SINU)/X0**4+P           *SINI*SINI*SIN2U/X0**5)
      DI=-(1.5D0*FMU*FJ2*RSQ*SIN2I*SINU*COSU)/(FN*A*A*DSQRT(1.D0-E*E)
     1   *X0**3)
      DOM=DI*TANU/(SINI)
      DL = M * DOM * COSI -1.5D0 * FMU * FJ2 * DSQRT( 1.D0 - E*E )* RSQ
     1     / FN / A *( SINU *( 1.D0 -3.D0 * SINI * SINI*SINU * SINU )/
     2     X0**4 +( X0* L / P  +( 1 D0 + X0 / P )* COSU )* SINI * SINI *
     3     SIN2U / X0**4 )
      DM = -L * DOM * COSI +1.5D0 * FMU * FJ2 * RSQ *DSQRT( 1.D0 - E*E )
     1    / FN / A *( COSU *( 1.D0 -3.D0 * SINI * SINI * SINU * SINU )/
     2    X0**4 -( X0* M / P  +( 1.D0 + X0 / P )* SINU )* SINI * SINI *
     3    SIN2U / X0**4 )
      DP = DA *( 1.D0 - L*L - M*M ) -2.D0 * A *( L*DL + M*DM )
      DU = FN * A * A * DSQRT( 1.D0 - E*E ) / X0**2 - DOM * COSI
      DX0 = ( DA *( 1.D0 - L*L - M*M ) -2.D0 * A *( L * DL + M * DM )) /
     $      DENOM - A *( 1.D0 - L*L - M*M )*( DL * COSU - L * DU * SINU
     &      + DM * SINU + M * DU * COSU ) / DENOM / DENOM
      IF(KPRINT.EQ.1)    WRITE(6,NAM3)
      WZ=DSQRT(FMU*P)/X0**2
      IF(KOMEGA.EQ.0) GO TO 30
      WX=DI/COSU
      WY=0.D0
      DDI=(-1.5D0*FJ2*RSQ*DSQRT(FMU/P)/X0**3)*((DA-A*DP/(2.D0*P)
     1   -3.D0*A*DX0/X0)*SIN2I*SINU*COSU+A*(DI*COS2I*SIN2U
     2   +DU*SIN2I*COS2U))
      DWX=(DDI*COSU+DI*DU*SINU)/COSU*COSU
      DWY=0.D0
      GO TO 40
   30 WX=0.D0
      WY=0.D0
      DWX=0.D0
      DWY=0.D0
   40 DWZ=WZ*(DP/(2.D0*P)-2.D0*DX0/X0)
      IF(KPRINT.EQ.1)   WRITE(6,NAM4)
      A2=2.D0*(WY*DZ-DY*WZ)+Z*DWY-Y*DWZ+WX*(Z*WZ+Y*WY)
      A3=-WZ*WZ-WY*WY+FMU/X0**3-(1.5D0*FMU*FJ2*RSQ/X0**5)
     1   *(1.D0-3.D0*SINI*SINI*SINU*SINU)
      B2=2.D0*(DX*WZ-DZ*WX)+X*DWZ-Z*DWX+WY*(Z*WZ+WX*(X+X0))
      B3=-WX*WX-WZ*WZ+FMU/X0**3-(1.5D0*FMU*FJ2*RSQ/X0**5)*SINI*SINI*
     $ COS2U
      C2=2.D0*(DY*WX-WY*(DX+DX0))+Y*DWX-X*DWY+WZ*(WY*Y+WX*X)-X0*DWY
      C3=-WY*WY-WX*WX+FMU/X0**3-(1.5D0*FMU*FJ2*RSQ/X0**5)*SINU*COS2I
      IF(KPRINT.EQ.1)    WRITE(6,NAM1)
      DDX=-A2-A3*X
      DDY=-B2-B3*Y
      DDZ=-C2-C3*Z-X0*WX*WZ
```

```
  100 DO 110 I=1,12                                                        00011900
  110 DELM(I)=DEL(I)                                                       00012100
      IF(KPRINT.EQ.1)  WRITE(6,NAM5)
      IF( KPRINT.EQ.0 )  GOTO 99
      WRITE(6,2002)
 2002 FORMAT('O EXIT' )
                                WRITE(6,2000) (ELEM(I),DELM(I),I=1,13)
   99 RETURN                                                               00012500
      END
```

```
      SUBROUTINE PLOT
      REAL*8          TIME,POS,VEL
      DIMENSION TIME(500),VEL(500),POS(500)
      DIMENSION T(500),R(500),V(500),Z(200)
      COMMON /PLOTS/ TIME,POS,VEL,J
C  CONVERT FROM DOUBLE PRECISION ARRAYS TO SINGLE PRECISION
      DO 10 I= 1,J
      T(I) = TIME(I)
      R(I) = POS(I)
   10 V(I) = VEL(I)
      CALL MODESG( Z,0 )
      CALL SETSMG( Z,84,'+' )
      CALL GRAPHG( Z,J,T,R,13,'TIME ( SECS )',15,'POSITION ( KM )',17,
     U 'TIME VS. POSITION' )
      CALL LINESG( Z,J,T,R )
      CALL PAGEG( Z,0,1,1 )
      CALL GRAPHG( Z,J,T,V,13,'TIME ( SECS )',19,'VELOCITY ( KM/SEC )',
     U 17,'TIME VS. VELOCITY' )
      CALL LINESG( Z,J,T,V )
      CALL PAGEG( Z,0,1,1 )
      CALL EXITG ( Z )
      RETURN
      END
```

```
      SUBROUTINE FNOL2(J,N,G,L,M,NE,X,Y,D)
      IMPLICIT REAL*8(A-H,O-Z,$)
C
C
      DOUBLE PRECISION XD,YD,YA,YC,YP,Y1
      DIMENSION Y(50),D(50),YB(30,6),GI2(30),GI3(30),GI4(30),EF(30),
     1EF1(30),EF2(30),EF3(30),Y1(30),ERROR(30),HA(30),YA(50),DA(50),
     2YC(30),YP(30),YD(50)
      EC=Y(N+3)
    1 H=G
    2 HZ=H
    3 LN=N+MAXO(L,3)
C     SUBROUTINE FNOL2(J,N,G,L,M,NE,X,Y,D,DERIV,TERM,OUT)
    4 NA=0
    5 NB=1
    6 NF=0
    7 NG=0
    8 F=0.D0
    9 FA=0.D0
   10 FB=0.D0
   11 FC=0.D0
   12 CONTINUE
   13 ENE=NE
C     DO 200 I=1,LN
C 200 YD(I)=DBLE(Y(I))
C     XD=DBLE(X)
      DO 200 I=1,LN
  200 YD(I)=Y(I)
      XD=X
   14 IF(J-3)15,21,15
   15 IF(NE)18,16,18
   16 JA=4
   17 GO TO 22
   18 RE1=10.D0**(-ENE)
   19 RE2=10.D0**(-ENE-3.0D0)
   20 REM=10.D0**(-ENE-1.5D0)
   21 JA=1
   22 DO 25 I=1,N
   23 DO 24 IC=1,5
   24 YB(I,IC)=0.D0
   25 ERROR(I)=0.D0
   26 CALL DERIV(X,Y,D)
      CALL TERM(X,Y,D,F)
      IF(DABS(F)-1.0D-9) 731,731,5209
 5209 CONTINUE
      DO 300 I=1,N
      GI2(I)=D(I)
      GI3(I)=D(I)
      GI4(I)=D(I)
  300 EF(I)=D(I)
C  27 CALL OUTPUT(X,Y,D,ERROR,N,L,H)
 27   CALL OUT(X,Y,D,ERROR,N,L,H)
   28 FD=Y(N+1)
   29 IF(J-2) 30,129,30
   30 GO TO(31,37,35,37),JA
   31 DO 33 I=1,LN
   32 YA(I)=YD(I)
   33 DA(I)=D(I)
   34 GO TO 37
   35 HB=H
   36 H=2.D0*H
   37 HD2 = .5D0*H
      DO 39 I=1,N
   38 YB(I,NB)=D(I)
      XL = D(I) * HD2
C  39 Y(I)=SNGL(YD(I)+XL)
```

```
   39  Y(I)=YD(I)+XL
C      X=SNGL(XD+HD2)
       X=XD+HD2
    40 CALL DERIV (X,Y,GI2)
    41 DO 42 I=1,N
       XL = GI2(I)*HD2
C   42 Y(I)=SNGL(YD(I)+XL)
  42   Y(I)=YD(I)+XL
    43 CALL DERIV (X,Y,GI3)
    44 DO 45 I=1,N
       XL=GI3(I)*H
C   45 Y(I)=SNGL(YD(I)+XL)
  45   Y(I)=YD(I)+XL
C      X=SNGL(XD+H)
       X=XD+H
    46 CALL DERIV(X,Y,GI4)
    47 HD6 =H/6.D0
       GO TO(48,55,60,66),JA
    48 DO 52 I=1,N
       XL=(D(I) + 2.D0*(GI2(I) + GI3(I)) +GI4(I))*HD6
    49 YC(I)=YD(I)+XL
    51 YD(I)=YA(I)
    52 ERROR(I)=0.D0
    53 JA=3
    54 GO TO 35
    55 DO 57 I=1,N
       XL=(D(I) + 2.D0*(GI2(I) + GI3(I)) +GI4(I))*HD6
    56 YD(I)=YD(I)+XL
C   57 ERROR(I)=SNGL(YD(I)-YP(I))/15.
  57   ERROR(I)=(YD(I)-YP(I))/15.D0
    58 JA=1
    59 GO TO 681
    60 DO 62 I=1,N
    61 YD(I)=YC(I)
       XL=(D(I) + 2.D0*(GI2(I) + GI3(I)) +GI4(I))*HD6
    62 YP(I)=YA(I)+XL
    63 H=HB
    64 JA=2
    65 GO TO 681
    66 DO 68 I=1,N
       XL=(D(I) + 2.D0*(GI2(I) + GI3(I)) +GI4(I))*HD6
    67 YD(I)=YD(I)+XL
    68 ERROR(I)=0.D0
   681 DO 69 I=1,N
C   69 Y(I)=SNGL(YD(I))
  69   Y(I)=YD(I)
       XD=XD+H
C      X=SNGL(XD)
       X=XD
    70 CALL DERIV(X,Y,D)
    71 FC=F
    72 CALL TERM(X,Y,D,F)
    73 IF(DABS(F)-1.0D-9 )731,731,733
   731 NF=5
   732 GO TO 124
   733 IF(F)74,124,76
    74 FA=1.D0
    75 GO TO 77
    76 FB=1.D0
    77 IF(FA-FB)83,78,83
    78 NF=NF+1
    79 JA=4
    80 NB=1
    81 H=H*F/(FC-F)
    82 IF(NF-4)37,37,124
    83 IF(NE)84,117,84
```

```
   84 IF(JA-1)117,85,117
   85 IF(J-3)86,117,86
   86 DO 95 I=1,N
      IF(Y(I))886,885,886
  885 HA(I)=1000.D0
      GO TO 95
  886 IF(EC)880,890,87
   87 IF(DABS(Y(I))-EC) 880,880,890
  880 IF(DABS(ERROR(I))-RE2) 882,94,881
  881 IF(DABS(ERROR(I))-RE1)94,94,882
  882 HA(I)=H*(REM/(DABS(ERROR(I))+.0000000001D0))**(.2D0)
  883 GO TO 95
  890 IF(DABS(ERROR(I)/Y(I))-RE2)892,94,891
  891 IF(DABS(ERROR(I)/Y(I))-RE1)94,94,892
  892 HA(I)=H*(REM/(DABS(ERROR(I)/Y(I))+.0000000001D0))**(.2D0)
  893 GO TO 95
   94 HA(I)=H
   95 CONTINUE
96      HB=DABS(HA(N))
97      DO 98 I=1,N
C 98    HB=AMIN1(ABS(HA(I)),HB)
98      HB=DMIN1(DABS(HA(I)),HB)
99      IF(DABS(H)-HB)100,117,101
100     IF(DABS(HZ)-DABS(H))101,101,116
101     DO 103 I=1,LN
102     YD(I)=YA(I)
C       Y(I)=SNGL(YD(I))
        Y(I)=YD(I)
103     D(I)=DA(I)
104     IF(NB-6) 107,105,105
105     XD=XD-H
106     GO TO 109
107     XD=XD-2.D0*H
108     HZ = H
109     H=DSIGN(HB,H)
        X=XD
C       X=SNGL(XD)
        CALL DERIV(X,Y,D)
        CALL TERM(X,Y,D,F)
  110 NB=1
  111 XABS=DABS(.000001D0*X)
  112 IF(DABS(H)-XABS)113,113,117
  113 NG=NG+1
114     H=DSIGN(XABS,H)
 115    IF(NG - 10)124,150,150
 150    WRITE(6,1261) H
1261    FORMAT (1H1,107HEXECUTION TERMINATED BECAUSE INTERVAL OF INTEGRATI
       1ON LESS THAN 1.0E -6 TIMES INDEPENDENT VARIABLE (X).  X =,1PD15.7)
        STOP
  116 HZ=H
  117 IF(M)118,118,121
  118 IF(     (Y(N+1)-FD)-Y(N+2))29,119,119
  119 FD=FD+Y(N+2)
  120 GO TO 124
  121 NA=NA+1
  122 IF(M-NA)123,123,29
  123 NA=0
C124    CALL OUTPUT(X,Y,D,ERROR,N,L,H)
 124    CALL OUT(X,Y,D,ERROR,N,L,H)
  125 IF(NF-4)29,29,126
  126 WRITE (6,127)
  127 FORMAT(1H0)
  128 RETURN
  129 NB=NB+1
  130 IF(NB-6)30,131,136
  131 DO 134 I=1,N
```

```
  132 EF3(I)=YB(I,3)
  133 EF2(I)=YB(I,4)
  134 EF1(I)=YB(I,5)
  135 GO TO 137
  136 NB=10
  137 HD24 =H/24.DO
      DO 138 I=1,N
      XL =(55.DO*D(I) -59.DO*EF1(I) +37.DO*EF2(I) -9.DO*EF3(I))*HD24
      YP(I)=YD(I)+XL
C 138 Y(I)=SNGL(YP(I))
C     X=SNGL(XD+H)
 138  Y(I)=YP(I)
      X=XD+H
  139 CALL DERIV(X,Y,EF)
  140 DO 142 I=1,LN
  141 YA(I)=YD(I)
  142 DA(I)=D(I)
  143 DO 148 I=1,N
      XL = (9.DO*EF(I) +19.DO*D(I) -5.DO*EF1(I) +EF2(I))*HD24
  144 YD(I)=YD(I)+XL
C 145 ERROR(I)=-SNGL(YD(I)-YP(I))/14.
 145  ERROR(I)=-(YD(I)-YP(I))/14.DO
  146 EF3(I)=EF2(I)
  147 EF2(I)=EF1(I)
  148 EF1(I)=D(I)
999   CONTINUE
  149 GO TO 681
       END
```

```
SUBROUTINE TERM( T,ELEM,DELM,COND )
IMPLICIT REAL*8(A-H,O-Z,$)
COMMON /CONST/ TF,STEP,TI,INTP,KUTMOL,EPOCH,FMU,FJ2,RSQ
DIMENSION ELEM(1), DELM(1)
COND = TF - T
RETURN
END
```

```
      SUBROUTINE OUT    (X,Y,D,ERROR,N,L,H)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION Y(9),D(9),ERROR(2)
      DIMENSION R(5000), V(5000), T(5000)
      DIMENSION DEG(5)
      COMMON /FLAG/ XO, KOMEGA,KPRINT,KPLOT
      COMMON /PLOTS/ T,R,V,J
      DATA TPI/ 6.2831853072 /
      VMAG(X,Y,Z) = DSQRT( X*X + Y*Y + Z*Z )
      DEGREE(X) = X * 57.2957795130823
      J = J + 1
      R(J) = VMAG( Y(1),Y(2),Y(3) )
      V(J)=VMAG( D(1),D(2),D(3) )
      T(J) = X
      WRITE(6,3)
    3 FORMAT(130(1H-))
      WRITE(6,4)  R(J),V(J)
    4 FORMAT('0',25X,'POSITION = ',D20.10,15X,'VELOCITY = ',D20.10 )
      E = VMAG( Y(5),Y(8),0.DO )
      IF( DABS(Y(5)).LE.1.D-30 .AND. DABS(Y(8)).LE.1.D-30 )   GOTO 50
      W = DATAN2( Y(8),Y(5) )
      GOTO 60
   50 W = 0.DO
   60 CONTINUE
      V2 = Y(9) - W
      V2 = DMOD( V2,TPI )
      W = DMOD( W,TPI )
      Y(6) = DMOD( Y(6),TPI )
      Y(7) = DMOD( Y(7),TPI )
      Y(9) = DMOD( Y(9),TPI )
      DEG(1) = DEGREE( Y( 7) )
      DEG(2) = DEGREE( W )
      DEG(3) = DEGREE( Y( 9) )
      DEG(4) = DEGREE( Y(  6))
      DEG(5) = DEGREE( V2 )
      WRITE(6,1001) X,(Y(I),I=1,3)
 1001 FORMAT('-',4X,'TIME (SEC) = ',D15.8,8X,'X (KM) = ',D15.8,8X,
     1 'Y (KM) = ',D15.8,8X,'Z (KM) = ',D15.8 )
      WRITE(6,1002)    XO,(D(I),I=1,3)
 1002 FORMAT('0',7X,'XO (KM) = ',D15.8,3X,'X'' (KM/SEC) = ',D15.8,3X,
     2 'Y'' (KM/SEC) = ',D15.8,3X,'Z'' (KM/SEC) = ',D15.8)
      WRITE(6,1003)  Y(4),E,Y(6),DEG(4)
 1003 FORMAT('0',8X,'A (KM) = ',D15.8,9X,'ECCEN = ',D15.8,10X,'INCL = '
     3 ,D15.8,' RAD = ',D15.8,' DEG')
      WRITE(6,1004) Y(7),DEG(1),  W  ,DEG(2)
 1004 FORMAT('0',8X,'L.A.N. = ',D15.8,' RAD = ',D15.8,' DEG',10X,
     4 'ARG OF PER = ',D15.8,' RAD = ',D15.8,' DEG')
      WRITE(6,1005)  V2,DEG(5),Y(9),DEG(3)
 1005 FORMAT('0',13X,'V = ',D15.8,' RAD = ',D15.8,' DEG',19X,'U = ',
     5 D15.8,' RAD = ',D15.8,' DEG')
      WRITE(6,1006) Y(5),Y(8)
 1006 FORMAT('0',13X,'L = 'D15.8,45X,'M = 'D15.8)
      RETURN
      END
```

```
BLOCK DATA
IMPLICIT REAL*8(A-H,O-Z,$)
REAL*8 I
LOGICAL KPLOT
DIMENSION R(5000), V(5000), T(5000)
COMMON /FLAG/ XO, KOMEGA,KPRINT,KPLOT
COMMON /INCOND/ A,E,I,OMEGA,W,U ,X,Y,Z,XDOT,YDOT,ZDOT
COMMON /CONST/ TF,STEP,TI,INTP,KUTMOL,EPOCH,FMU,FJ2,RSQ
COMMON /PLOTS/ T,R,V,J
DATA FMU,RSQ      / 3.986032D5, 4.068098D7/
DATA T,R,V,J / 15000*0.D0,0 /
DATA A,E,I,OMEGA,U ,W/ 1.5D4,1.D-2,3*45.D0,0.D0 /
DATA STEP,TI,TF,INTP,KUTMOL/ 6.D2,0.D0,8.64D4,1,2 /
DATA EPOCH ] 720101.D0 /
DATA X,Y,Z / 3*1.D0 /
DATA XDOT,YDOT,ZDOT / 3*1.D-1/
DATA    FJ2 / 1.0823D-3 /
DATA KOMEGA,KPRINT,KPLOT / 1,0,.TRUE. /
END
```

APPENDIX V

APPENDIX V

```
C   TWO SATELLITE MOTION STUDY - ANALYTIC THEORY
C       LAST REVISION 10-16-70
C
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8  I
      LOGICAL PRINT
      DIMENSION TYPE(5,3)
      DIMENSION ZERO(4),ELEM(20),DELM(20)
      COMMON / ELMENT / A,E,I,OM,V,U,WO
      COMMON /INTEG/ TF,TINCR,TI,INTP,KUTMOL
      COMMON / CONST / FJ2,FMU,RSQ,ZERO
      COMMON /FLAG/ PRINT,J,NEQ
      COMMON /ANS/ X,Y,XO,DX,DY
      RADIAN(X) = X / 57.2957795130823
      DATA TYPE /'RUNGE-KU','TTA THRO','UGHOUT  ',2*'        ',
     1 'RUNGE-KU','TTA AND ','ADAMS-MO','ULTON   ','        ',
     2 'RUNGE-KU','TTA WITH',' ERROR C','ALCULATI','ON      '/
       NAMELIST /INIT/ ZERO,A,E,I,V,OM,WO,FJ2
      NAMELIST /INTEG/ TF,TINCR,TI,INTP,KUTMOL,PRINT,NEQ
      READ(5,INIT)
      READ(5,INTEG)
      WRITE (6,1)
      WRITE(6,1001)
 1001 FORMAT( 25(  /),35X,'ANALYTIC SOLUTION OF TWO-SATELLITE MOTION '
     F ,' STUDY')
      WRITE(6,1)
    1 FORMAT('1')
      WRITE(6,5)
    5 FORMAT( 30X,10(1H*),10X,'INITIAL CONDITIONS',10X,10(1H*) )
      WRITE(6,8) A,E,I,OM,WO  ,V,FJ2,ZERO(1),ZERO(3),ZERO(2),ZERO(4)
    8 FORMAT( /39X,'SEMI-MAJOR AXIS (A) = ',D20.10/42X,'ECCENTRICITY (E)
     1 = ',D20.10/42X,'INCLINATION ( I) = ',D20.10/26X,'LONGITUDE OF',
     2 ' ASCENDING NODF (OM) = ',D20.10/34X,'ARGUMENT OF PERIGEE (WO) ='
     3 D21.10/   42X,'TRUE ANOMALY (V) = ',D20.10/42X,'OBLATENESS (J20)
     4= ',D20.10/57X,'X ='D20.10/57X,'Y ='D20.10/56X,'X'' ='D20.10/
     5 56X,'Y'' =',D20.10 )
      WRITE(6,7) (TYPE(J,KUTMOL),J=1,5),TINCR,INTP,NEQ
    7 FORMAT( ////48X,'INTEGRATION PARAMETERS'/48X,22(1H-)//45X,
     1 'METHOD   ',5A8/45X,'STEPSIZE   ',D20.10/45X,'PRINT FREQUENCY
     2 ,I8/45X,'NUMBER OF EQUATIONS = 'I2)
      WRITE(6,1)
      U = V + WO
      I = RADIAN(I)
      OM = RADIAN(OM)
      U = RADIAN(U)
      V = RADIAN(V)
      WO = RADIAN(WO)
      ELEM( 1) = A
      ELEM( 2) =   E
      ELEM( 3) = I
      ELEM( 4) = OM
      ELEM( 5) = U
      ELEM(6) = WO
      X = ZERO(1)
      Y = ZERO(3)
      DX = ZERO(2)
      DY = ZERO(4)
      XO = A *( 1.D0 - E * E )/( 1.D0 + E * DCOS(V) )
      LL = 0
      NE = 0
```

PRECEDING PAGE BLANK NOT FILMED

```
      CALL FNOL2( KUTMOL,NEQ,TINCR,LL,INTP,NE,TI,ELEM,DELM)
      WRITE(6,9)
    9 FORMAT('-END OF RUN')
   99 RETURN
      END
```

```
      SUBROUTINE DERIV( T,EL,DEL )
      IMPLICIT REAL*8(A-H,O-Z)
      REAL*8 N,NBAR,NPN,NMN,NPO,I,NSS
      REAL*8 LAM1,LAM2,LAM3,LAM4,LAM5,LAM6,LAM7,LAM8,LAM9,LAM10,
     L        LAM11,LAM12
      LOGICAL PRINT
      DIMENSION ELEM(200),DELM(200),EL(200),DEL(200),ZERO(4)
      EQUIVALENCE (ELEM( 1),A), (ELEM( 2), E), (ELEM( 3),I),
     E            (ELEM( 4),OM), (ELEM( 5),U), (ELEM( 6),WO)
      EQUIVALENCE (DELM( 1),DA),  (DELM( 2),DE ), (DELM( 3),DI),
     E            (DELM( 4),DOM), (DELM( 5),DU), (DELM( 6),DWO)
      COMMON / CONST / FJ2,FMU,RSQ,ZERO
      COMMON /FLAG/ PRINT,J,NEQ
      COMMON/ ANS / X,Y,XO,DX,DY
      COMMON / ALFA / ALFO,ALF1,ALF2,ALF3
      NAMELIST /TRIG/ COSV,SINV,COSU,SINU,COSI,SINI,COS2U,SIN2U,SIN2I,
     U COS2I,TANU,COTI
      NAMELIST /NAM2/ V,U, I,P,XO, N
      NAMELIST /NAM3/ DA,DE,DP,C,DV,DWO,DI,DOM
      NAMELIST /NAM4/ XOO,XO1,X10
      NAMELIST /NAM5 / YOO,YO1,Y10
      NAMELIST /NAM6/ N,DOCI,NBAR
      NAMELIST /NAM1/ DXOO,DXO1,DX10
      NAMELIST /NAM7/ DYOO,DYO1,DY10
      NAMELIST / LAMBDA / LAM1,LAM2,LAM3,LAM4,LAM5,LAM6,LAM7,LAM8,
     N                    LAM9,LAM10,LAM11,LAM12
      NAMELIST / COEF1 / A1,B1,C1,D1,ALF3D,ALF3N
      NAMELIST / COEF2 / A2,B2,C2,D2
      NAMELIST/ TIMEO /  TXO,TYO,TDXO,TDYO
C
C    SUBROUTINE DERIV COMPUTES THE DERIVATIVES OF THE ELEMENTS ARRAY.
C
C    INPUT
C        T       - TIME
C        ELEM    - ARRAY OF ELEMENTS
C
C    OUTPUT
C        DELM    - ARRAY OF DERIVATIVES OF ELEMENTS
C
      DO 100 L=1,NEQ
      ELEM(L) = EL(L)
      DELM(L) = DEL(L)
  100 CONTINUE
      I = DMOD(I,3.6D2)
      WO = DMOD(WO,3.6D2)
      IF( .NOT.PRINT )  GOTO 500
C     WRITE(6,2001)
C2001 FORMAT('0 ENTER' )
      WRITE(6,2000) ( ELEM(K),DELM(K),K=1,NEQ)
 2000 FORMAT( 22X,'ELEMENTS',9X,'DERIVATIVES'//(10X,2D20.8))
  500 CONTINUE
C   COMPUTE FREQUENTLY USED TRIGONOMETRIC FUNCTIONS
      SINI = DSIN(I)
      SIN2I=DSIN(2.D0* I)
      COSI = DCOS(I)
      COS2I=DCOS(2.D0* I)
      COTI=DCOTAN( I)
      P=A*(1.D0-E*E)
      N = DSQRT( FMU / A / A / A )
      DWO =   .75D0 *N *FJ2 *RSQ /P /P *( 5.D0 * COSI**2 -1.D0 )
   20 V = U - WO
     XO = P / ( 1.D0 + E * DCOS(V) )
     RDA = RSQ /A /A
     NPO = N + WO
     EPS = 1.5D0 * FJ2
     COSV=DCOS(V)
```

```
      SINV=DSIN(V)
      COSU=DCOS(U)
      COS2U=DCOS(2.D0*U)
      SINU=DSIN(U)
      SIN2U=DSIN(2.D0*U)
      TANU=DTAN(U)
      IF( PRINT )  WRITE(6,TRIG)
      C = COSV *( 1.D0 -3.D0 * SINI * SINI * SINU * SINU )/ XO**4
C     -( 1.D0 + XO / P )* SINI * SINI * SIN2U * SINV / XO**4
      IF( PRINT )   WRITE(6,NAM2 )
C
      DA = -3.D0 * FMU * FJ2 * RSQ / XO**4 /( N * DSQRT( 1.D0 - E*E ))
     1      *(( 1.D0 -3.D0 * SINI*SINI*SINU*SINU ) * E * DSIN(V) +
     2      P / XO * SINI * SINI * SIN2U )

      DE = -EPS * FMU * RSQ / XO**4 * DSQRT( 1.D0 -E*E ) / N / A *((
     1      1.D0 -3.D0 * SINI*SINI * SINU*SINU )* DSIN(V) + XO / P *( E
     2      * DCOS(V) *( DCOS(V) +2.D0 ) + E )* SINI*SINI * SIN2U )
      DI = -EPS * FMU * RDA * SIN2I    *      SINU    *   COSU    / N /
     D DSQRT( 1.D0 - E*E ) / XO / XO / XO
      DOM = DI *  TANU   / SINI
      DU =  N * A * A * DSQRT( 1.D0 - E*E ) / XO**2 - DOM * COSI
      NBAR = N + DOM * COSI
      DOCI = DOM * COSI
      WRITE(6,NAM6)
      NPN = NBAR + N
      NMN = NBAR - N
      TNPN = 2.D0 * NBAR + N
      TNMN = 2.D0 * NBAR - N
      NSS = NBAR * NBAR * SINI * SINI
      OZT = RDA * NBAR *( 1.D0 -1.5D0 * SINI   * SINI ) / 8.D0
      IF( PRINT )   WRITE(6,NAM3)
C
      A1 = 0.5D0 / NBAR+ E / 2.D0 * NBAR / N *( N-4.D0 * NBAR )/ TNPN /
     1    TNMN +3.D0 / 16.D0 * RDA * NSS / NBAR * EPS /
     2    (NBAR**2 - NPO**2) + EPS * OZT / 2.D0 / NBAR / NBAR
      B1 = 1.D0 + E    * NBAR / N *( 4.D0 *NBAR**2 + N*NBAR -8.D0 *N*N )
     1    /( 4.D0 *NBAR**2 - N*N ) + EPS * RDA * NSS / 16.D0
     2    / NPO                     *(-( 3.D0 *NPO +5.D0 *NBAR )/(NPN+WO)
     3    /( TNPN +WO ) + 3.D0 *NPO -5.D0 *NBAR /( NPO-NBAR )/(TNPN-WO))
      C1 = 1.D0 + E *  NBAR *((NBAR +8.D0 * N )/ TNPN +( 3.D0 * NBAR
     1    -8.D0 * N )/ TNMN )/ 2.D0 / N + EPS *( OZT/ NBAR - RDA * NBAR
     2                    * NSS *( 3.D0 * NPO + 5.D0 * NBAR )/ 16.D0 /
     3    NPO /( NPN + WO )**2 /( TNPN + WO ) - RDA / 8.D0 * NSS
     4                   / ( NPN + WO )**2 + RDA * NSS * NBAR
     5          *( -3.D0 * NPO +5.D0 * NBAR ) / 16.D0 / NPO /( WO-NMN)**2
     6    /( TNMN - WO ) - RDA * NSS / 8.D0 /
     7    ( WO - NMN )**2 )
      D1 = 1.D0 -E * NBAR**2 /N *( 4.D0 * NBAR - N )/ TNMN / TNPN + EPS
     1    * RDA * NSS / 2.D0 / NPO / NPO *( NBAR * NBAR / 2.D0 / ( NBAR
     2    * NBAR - NPO * NPO ) - 1.D0 )
      A2 = 2.D0 * NBAR *( 1.D0 + E * NBAR / N ) - EPS * RDA * NSS / 8.D0
     1    / NPO *((3.D0 * NPO +5.D0 * NBAR )/( TNPN + WO ) +( 3.D0 * NPO
     2    -5.D0 * NBAR )/( TNMN - WO ))
      B2 = E * 2.D0 * NBAR * NBAR * NMN / TNMN / TNPN + EPS * RDA * NSS
     1    * NBAR / 2.D0 / ( NBAR * NBAR - NPO * NPO )
      C2 = -E * NBAR * NMN / TNMN / TNPN -EPS *( 0.375D0 * RDA * NSS /
     1    ( NBAR * NBAR - NPO * NPO ) + OZT/ NBAR )
      D2 = -2.D0 * NBAR *( 1.D0 + E * NBAR / N ) + EPS *( -2.D0 * OZT+
     1    RDA * NSS *(( 3.D0 * NPO +5.D0 * NBAR)/ ( NPN + WO ) * NBAR /
     2    NPO /(TNPN + WO )            - ( 3.D0 * NPO -5.D0 * NBAR )/( NMN
     3    + WO ) * NBAR / NPO / ( TNMN - WO )) / 8.D0   + RDA * NSS /4.D0
     4    *( 1.D0 /( NPN + WO )- ( WO - NMN )))
      ALF3N = C1 * ZERO(2) - A2 * ZERO(3)
      ALF3D = B2 * C1 - A2 * D1
C
```

```
C      ALPHA TERMS
C
        ALFO = ( D2 * ZERO(1) -B1 * ZERO(4) )/( A1 * D2 - B1 * C2 )
        ALF1 = ( B2 * ZERO(3) -D1 * ZERO(2) )/( B2 * C1 - A2 * D1 )
        ALF2 = ( A1 * ZERO(4) -C2 * ZERO(1) )/( A1 * D2 - B1 * C2 )
        ALF3 = ( C1 * ZERO(2) -A2 * ZERO(3) )/( B2 * C1 - A2 * D1 )
C
C     LAMBDA TERMS
C
        LAM1 = -( NBAR**2 +8.DO *N *NBAR )/ 2.DO *( 4.DO * NBAR + N )/
      1       TNPN * ALF1
        LAM2 = ( 3.DO * NBAR**2 -8.DO *N *NBAR )/ 2.DO *( 4.DO * NBAR -N)
      1       / TNMN * ALF1
        LAM3 = -( NBAR**2 +8.DO *N *NBAR )/ 2.DO *( 4.DO * NBAR + N )/
      1       TNPN * ALF2
        LAM4 = ( 3.DO * NBAR**2 -8.DO *N *NBAR )/ 2.DO *( 4.DO * NBAR -N)
      1       / TNMN * ALF2
        LAM5 = -NBAR / N * ( 4.DO * NBAR - N )/ 2.DO * ALFO
        LAM6 = 2.DO * NBAR**2 / N * NMN * ALF3
        LAM7 = RDA * NSS              * NBAR / NPO * ALF3
        LAM8 = .75DO * RDA * NSS                / NBAR * ALFO
        LAM9 = RDA * NSS              *( 3.DO * NPO +5.DO * NBAR )/ 4.DO
      1       / (WO + NPN ) * ALF1
        LAM10 = -RDA * NSS              *( 3.DO * NPO -5.DO * NBAR ) /
      1       4.DO / ( WO - NMN ) * ALF1
        LAM11 = RDA * NSS               *( 3.DO * NPO +5.DO * NBAR ) /
      1       4.DO / ( WO + NPN ) * ALF2
        LAM12 = RDA * NSS               *( 3.DO * NPO -5.DO * NBAR ) /
      1       4.DO / ( WO - NMN ) * ALF2
        IF( T.GT. 0.DO )  GOTO 300
        X = ZERO(1)
        Y = ZERO(3)
        DX = ZERO(2)
        DY = ZERO(4)
        IF( .NOT.PRINT )   GOTO 400
        TXO = ALFO * A1 + ALF2 * B1
        TYO = ALF1 * C1 + ALF3 * D1
        TDXO = ALF1 * A2 + ALF3 * B2
        TDYO = ALFO * C2 + ALF2 * D2
        WRITE(6,TIMEO)
        GOTO 400
  300 CONTINUE
        XOO = ALF1 * DSIN( 2.DO*NBAR*T ) +ALF2 * DCOS( 2.DO * NBAR * T )
      1      + ALFO / 2.DO / NBAR
        XO1 = LAM6 / TNMN / TNPN * DSIN( N*T ) + LAM5 / TNMN / TNPN *
      1      DCOS( N*T ) - LAM1 / N /( 4.DO *NBAR +N ) * DSIN( TNPN*T ) +
      2      LAM2 / N /( 4.DO *NBAR -N ) * DSIN( TNMN*T ) -LAM3 / N /
      3      ( 4.DO *NBAR + N ) * DCOS( TNPN*T ) + LAM4 / N / ( 4.DO *NBAR
      4      -N)*DCOS( TNMN *T )
        X1O = ( LAM7 * DSIN( 2.DO * NPO * T ) + LAM8 * DCOS( 2.DO * NPO *
      1      T )) / 4.DO / ( NBAR * NBAR - NPO * NPO ) - ( LAM9 * DSIN(
      2      2.DO *( NPN + WO )* T ) + LAM11 * DCOS( 2.DO *( NPN + WO )* T
      3      )) / 4.DO / NPO / ( TNPN + WO ) + ( LAM10 * DSIN( 2.DO *( WO
      4      - NMN )* T ) + LAM12 * DCOS( 2.DO *( WO - NMN )* T ))/ 4.DO
      5      / NPO /( TNMN - WO ) + .5DO * OZT * ALFO / NBAR / NBAR
        YOO = ALF1 * DCOS( 2.DO * NBAR * T ) - ALF2 * DSIN( 2.DO * NBAR *
      Y      T ) + ALF3
        Y1O =    OZT / NBAR *( ALF1 * DCOS( 2.DO * NBAR * T ) - ALF2 * DSIN
      1      ( 2.DO * NBAR * T )) +   NBAR / 4.DO / NPO /( NBAR * NBAR -
      2      NPO * NPO ) *( LAM7 * DCOS( 2.DO * NPO * T ) - LAM8 * DSIN(
      3      2.DO * NPO * T )) - RDA * NSS * ALF3 / 2.DO / NPO / NPO *
      4      DCOS( 2.DO * NPO * T ) +  NBAR / NPO / ( NPN + WO ) / 4.DO /
      5      ( TNPN + WO )*( LAM11 * DSIN( 2.DO *( NPN + WO)*T)- LAM9 *
      6      DCOS( 2.DO *( NPN + WO ))) +  RDA * NSS / 8.DO *((ALF2 * DSIN
      7      ( 2.DO *( NPN + WO )*T) - ALF1 * DCOS( 2.DO *( NPN + WO )* T
      8      ))/( NPN + WO )**2  -                  ( ALF1 * DCOS( 2.DO *
```

```
     9      ( WO - NMN )* T ) + ALF2 * DSIN( 2.DO *( WO - NMN )* T )) /
     A      ( WO - NMN )**2 ) + NBAR / NPO / 4.DO /( WO - NMN ) /( TNMN
     B      - WO )*( LAM10 * DCOS( 2.DO *( WO - NMN )* T ) - LAM12 * DSIN
     C      ( 2.DO *( WO - NMN )* T ))  - OZT * ALFO * T / NBAR
      Y01 = - NBAR / N *( ALFO / N * DSIN( N*T ) + NBAR / N * ALF3 *
     1      DCOS( N*T ) + 2.DO / TNPN / TNMN *( LAM5 * DSIN( N*T ) - LAM6
     2      * DCOS( N*T ))+  ( 4.DO * NBAR / N /( TNPN +2.DO * NBAR ) *
     3      ( LAM3 * DSIN( TNPN*T ) - LAM1 * DCOS( TNPN*T )) -    NBAR *
     4      ( NBAR +8.DO * N )/ TNPN *( ALF2 * DSIN( TNPN*T ) - ALF1 *
     5      DCOS( TNPN*T ))) / 2.DO / TNPN  +( NBAR *( 3.DO * NBAR -8.DO
     6      * N )/ TNMN *( ALF2 * DSIN( TNMN*T ) - ALF1 * DCOS( TNMN*T ))
     7      - 4.DO * NBAR / N / ( TNMN +2.DO * NBAR)* ( LAM4 * DSIN( TNMN
     8      *T ) - LAM2 * DCOS( TNMN*T )))/ TNMN *.5DO
      DX00 = 2.DO * NBAR *(ALF1 * DCOS( 2.DO * NBAR * T ) - ALF2 *
     1       DSIN( 2.DO * NBAR * T ))
      DX01 = N / TNMN / TNPN *( LAM6 * DCOS( N*T ) - LAM5 * DSIN( N*T ))
     1       - TNPN / N /( 4.DO * NBAR + N )*( DCOS( TNPN * T ) * LAM1 -
     2       DSIN( TNPN * T )) * LAM3  + TNMN /  N /( 4.DO * NBAR - N )*
     3       ( LAM2 * DCOS( TNMN * T ) - LAM4 * DSIN( TNMN * T ))
      DX10 = 0.5DO * NPO /( NBAR*NBAR - NPO*NPO )*( LAM7 * DCOS( 2.DO *
     1       NPO *T ) -LAM8 * DSIN( 2.DO * NPO * T ))+ 0.5DO *( WO + NPN)
     2       / NPO /( WO + TNPN )*( LAM11 * DSIN(2.DO*( WO+ NPN )*T )
     3       - LAM9 * DCOS( 2.DO *( WO+NPN )*T )) +0.5DO *( WO - NMN ) /
     4       ( TNMN - WO )*( LAM10 * DCOS( 2.DO *( WO-NMN )*T ) - LAM12
     5       * DSIN( 2.DO *( WO-NMN )*T ))/ NPO
      DY00 = -2.DO * NBAR *( ALF1 * DSIN( 2.DO * NBAR * T ) + ALF2 *
     1       DCOS( 2.DO * NBAR * T ))
      DY01 = -NBAR *( ALFO / N +2.DO * LAM5 / TNMN / TNPN )* DCOS( N*T )
     1       +NBAR *( NBAR / N * ALF3 -2.DO * LAM6 / TNPN / TNMN ) *
     2       DSIN( N*T ) + NBAR * 0.5DO *( 4.DO * LAM3 / N /( 4.DO * NBAR
     3       + N ) -( NBAR +8.DO * N )/ TNPN * ALF2 )* DCOS( TNPN * T )
     4       + NBAR * .5DO *(( 3.DO * NBAR -8.DO * N )/ TNMN * ALF2 -4.DO
     5       * LAM4 / N /( 4.DO * NBAR - N ))* DCOS( TNMN*T ) + NBAR *
     6       0.5DO *( 4.DO * LAM1 / N /( 4.DO * NBAR + N ) -( NBAR +8.DO
     7       * N )/ TNPN * ALF1 )* DSIN( TNPN*T ) + NBAR * 0.5DO *(( 3.DO
     8       * NBAR -8.DO * N )* ALF1 / TNMN -4.DO * LAM2 / N /( 4.DO *
     9       NBAR - N ))* DSIN( TNMN*T )
      DY10 = -2.DO * OZT *( ALF2 * DCOS( 2.DO * NBAR * T ) + ALF1 *
     1       DSIN( 2.DO * NBAR * T )) - NBAR * LAM8 * 0.5DO /( NBAR*NBAR
     2       - NPO*NPO )* DCOS( 2.DO * NPO * T ) -( NBAR * LAM7 * 0.5DO
     3       /( NBAR*NBAR - NPO*NPO ) - RDA * NSS * ALF3 / NPO ) * DSIN(
     4       2.DO * NPO * T )+( LAM11 * NBAR * 0.5DO / NPO / ( TNPN + WO)
     5       +0.25DO * RDA * NSS * ALF2 /( NPN + WO))* DCOS( 2.DO *( NPN
     6       + WO)* T ) -( LAM12 * NBAR * 0.5DO / NPO /( TNMN - WO ) +
     7       .25DO * RDA * NSS * ALF2 /( WO - NMN ))* DCOS(2.DO *( WO-NMN
     8       )*T ) +( LAM9 * NBAR *0.5DO / NPO /( TNPN + WO )+ 0.25DO *
     9       RDA * NSS * ALF1 /( NPN + WO))* DSIN(2.DO *(NPN+WO)*T)-(
     A       LAM10 * NBAR *0.5DO / NPO /( TNMN-WO ) -0.25DO * RDA * NSS
     B       * ALF1 /(WO - NMN ))* DSIN(2.DO *(WO-NMN)*T)-OZT*ALFO / NBAR
      X = X00 + E * X01 + EPS * X10
      Y = Y00 + E * Y01 + EPS * Y10
      DX = DX00 + E * DX01 + EPS * DX10
      DY = DY00 + E * DY01 + EPS * DY10
  400 IF( .NOT.PRINT )  GOTO 600
      WRITE(6,COEF1)
      WRITE(6,COEF2)
      WRITE(6,NAM4)
      WRITE(6,NAM5)
      WRITE(6,NAM1)
      WRITE(6,NAM7)
      WRITE(6,LAMBDA)
  600 CONTINUE
      DO 200 L=1,NEQ
      DEL(L) = DELM(L)
  200 CONTINUE
   99 RETURN                                                            00012500
      END
```

```
      SUBROUTINE OUT( T,ELEM,D,ERROR,N,LK,H )
      IMPLICIT REAL*8(A-H,O-Z)
      REAL*8 I
      LOGICAL PRINT
      DIMENSION T(1),ELEM(1),D(1),ERROR(1)
      DIMENSION DEG(5),ELE(500)
      COMMON /FLAG/ PRINT,J,NEQ
      COMMON/ ANS / X,Y,XO,DX,DY
      COMMON / ALFA / ALFO,ALF1,ALF2,ALF3
      EQUIVALENCE (ELE ( 1),A), (ELE ( 2),E),  (ELE ( 3),I),
     E            (ELE ( 4),OM), (ELE ( 5),U), (ELE ( 6),WO)
      DATA TPI/ 6.2831853072 /
      DEGREE(X) = X * 57.2957795130823
      VMAG(X,Y) = DSQRT( X*X + Y*Y )
      J = J + 1
      WRITE(6,3)
    3 FORMAT(130(1H-))
      POS = VMAG(X,Y)
      DO 10 K=1,NEQ
      ELE(K) = ELEM(K)
   10 CONTINUE
      VEL = VMAG(DX,DY)
      WRITE(6,4) POS,VEL
    4 FORMAT('0',25X,'POSITION = ',D20.10,15X,'VELOCITY = ',D20.10 )
      V = U - WO
      DO 20 K=3,6
   20 ELE (K) = DMOD( ELE (K),TPI )
      V = DMOD( V,TPI )
      DEG(1) = DEGREE( ELE (3))
      DEG(2) = DEGREE( ELE (4))
      DEG(3) = DEGREE( ELE (6) )
      DEG(4) = DEGREE( ELE (5))
      DEG(5) = DEGREE( V )
      WRITE(6,1001) T,X,Y,XO
 1001 FORMAT('-',4X,'TIME (SEC) = ',D15.8,8X,'X (KM) = ',D15.8,8X,
     1 'Y (KM) = ',D15.8,8X,'XO (KM) = ',D15.8 )
      WRITE(6,1003) A,E,I,DEG(1)
 1003 FORMAT('0',8X,'A (KM) = ',D15.8,9X,'ECCEN = ',D15.8,10X,'INCL = '
     3 ,D15.8,' RAD = ',D15.8,' DEG')
      WRITE(6,1004) ELEM(4),DEG(2),ELEM(6),DEG(3)
 1004 FORMAT('0',8X,'L.A.N. = ',D15.8,' RAD = ',D15.8,' DEG',10X,
     4 'ARG OF PER = ',D15.8,' RAD = ',D15.8,' DEG')
      WRITE(6,1005)  V,DEG(5),U,DEG(4)
 1005 FORMAT('0',13X,'V = ',D15.8,' RAD = ',D15.8,' DEG',19X,'U = ',
     5 D15.8,' RAD = ',D15.8,' DEG')
      WRITE(6,1006) ALFO,ALF1,ALF2,ALF3
 1006 FORMAT('0',10X,'ALFO ='D15.8,11X,'ALF1 ='D15.8,11X,'ALF2 ='D15.8,
     $ 11X,'ALF3 ='D15.8)
   50 RETURN
      END
```

```
SUBROUTINE TERM( T,ELEM,DELM,COND )
IMPLICIT REAL*8(A-H,O-Z,$)
DIMENSION ELEM(1), DELM(1)
COMMON /INTEG/ TF,TINCR,TI,INTP,KUTMOL
COND = TF - T
RETURN
END
```

```
BLOCK DATA
IMPLICIT REAL*8(A-H,O-Z,$)
REAL*8 I
LOGICAL PRINT
DIMENSION ZERO(4)
COMMON / CONST / FJ2,FMU,RSQ,ZERO
COMMON /INTEG/ TF,STEP,TI,INTP,KUTMOL
COMMON / ELMENT / A,E,I,OM,V,U,WO
COMMON /FLAG/ PRINT,J,NEQ
DATA   FJ2 / 1.0823D-3 /
DATA FMU,RSQ     / 3.986032D5, 4.068098D7/
DATA ZERO / 4*1.D0 /
DATA STEP,TI,TF,INTP,KUTMOL/ 6.D2,0.D0,8.6404,1,2 /
DATA A,E,I,U,V / 1.5D0, 1.D-2, 45.D0, 2*0.0D0 /
DATA OM,WO / 2*45.D0 /
DATA PRINT /.FALSE./
END
```

```
//GO.DATA5 DD *
 &INIT
 FJ2=0.D0,
 A=2.0D4,
 E=1.D-4,
 I=10.D0,
 OM=20.D0,
 V=45.D0,
 WC=0.D0,
 ZERO=1.D0,1.D-3,1.D0,1.D-3
 &END
 &INTEG
 TF=6.048D5,
 NEQ=6,
 INTP=6,
 &END
```