Академия наук СССР
ИНСТИТУТ КОСМИЧЕСКИХ ИССЛЕДОВАНИЙ

Пр - 420

Б.Ц. Бахшиян, А.А. Суханов

УНИВЕРСАЛЬНЫЕ АЛГОРИТМЫ И ПРОГРАММЫ ВЫЧИСЛЕНИЯ ПАРАМЕТРОВ ДВИЖЕНИЯ В ЗАДАЧЕ ДВУХ ТЕЛ

Москва

АКАДЕМИЯ НАУК СССР

ИНСТИТУТ КОСМИЧЕСКИХ ИССЛЕДОВАНИЙ

Пр – 420

Б.Ц. Бахшиян, А.А. Суханов

УНИВЕРСАЛЬНЫЕ АЛГОРИТМЫ И ПРОГРАММЫ ВЫЧИСЛЕНИЯ ПАРАМЕТРОВ ДВИЖЕНИЯ В ЗАДАЧЕ ДВУХ ТЕЛ

1978

Приведены алгоритмы и программы на языке ФОРТРАН для вычисления фазовых координат, элементов орбиты, частных производных первого и второго порядка от одних параметров движения по другим в задаче двух тел. Указанные алгоритмы (большинство из которых разработано авторами) пригодны при любых значениях эксцентриситета и удобны для вычисления различных навигационных параметров орбиты.

The algorithms and FORTRAN-programs for computing positions and velocities, orbital elements and first and second partial derivatives in the Two-Body problem are presented. These algorithms (most of them are developed by the authors) are applicable for any value of eccentricity and are convenient for computing various navigation parameters.

СОДЕРЖАНИЕ

I. ВВЕДЕНИЕ

I.I. Предварительные замечания

В задачах космической навигации и небесной механики возникает необходимость в вычислении различных параметров оскулирующей кеплеровской орбиты: фазовых координат, элементов орбиты, матриц частных производных от одних параметров траектории по другим и т.п. Желательно, чтобы формулы для этих параметров были достаточно просты и универсальны, т.е. не имели бы неопределенностей и особенностей при переходе от одного типа кеплеровской орбиты к другой (за исключением тех случаев, когда эти особенности вытекают из самого свойства используемого параметра).

В настоящей работе приведены универсальные и наиболее эффективные с точки зрения авторов алгоритмы. Решение задачи двух тел и вычисление времени перелета между двумя положениями на орбите производятся согласно работе [1], а расчет изохронных производных первого и второго порядка от прямоугольных фазовых координат по их начальным значениям и гравитационному параметру – согласно работе авторов [2]. В п.3 переведены алгоритмы вычисления различных элементов орбиты и частных производных от прямоугольных фазовых координат по различным совокупностям из шести элементов орбиты. Обычно для производных по каждой совокупности элементов выводится свой набор формул. Авторами создан алгоритм (см. п.3), который позволяет объединить в единую систему различные наборы формул для 80 совокупностей элементов и с помощью одной программы вычислять производные по заданному набору элементов. Это достигнуто за счет введения основной совокупности элементов, пригодной для любых непрямолинейных орбит и перехода от этой совокупности к заданной. Такой подход обеспечивает достаточно простую реализацию вычисления на ЭВМ элементов и производных, а также дает возможность легко получать формулы для производных по любому набору элементов, не рассмотренному в данной работе.

В Приложении приведены описания программ, соответствующих изложенным алгоритмам, и тексты этих программ на языке ФОРТРАН.

1.2. Сводка обозначений и используемых параметров

Вводимые векторы рассматриваются в формулах как матрицы-столбцы. Индекс "T" означает транспонирование, например, $\bar{r}^T$. Производная от вектора по скаляру есть вектор, от скаляра по вектору – матрица-строка, производная от n-мерного вектора по m-мерному вектору – матрица размеров $n \times m$.

Перечислим теперь основные из используемых параметров движения:

μ – гравитационный параметр;

t, t_0 – текущий и начальный моменты времени;

$\bar{r}$, $\bar{v}$ – 3-мерные векторы координат и скоростей в инерциальной системе координат, начало которой совпадает с притягивающим центром;

$r = |\bar{r}|$, $v = |\bar{v}|$;

$\bar{x} = \begin{bmatrix} \bar{r} \\ \bar{v} \end{bmatrix}$ – 6-мерный вектор.

Для отличия значений параметров движения и других функций времени в момент t_0 от их значений в момент времени t

используется индекс "о"; например, $\bar{r}(t_0) = \bar{r}_0$ и т.п.

Для первых интегралов задачи двух тел (к которым относятся и элементы орбиты) используются следующие общепринятые обозначения:

$\bar{c} = \bar{r} \times \bar{v}$ - интеграл площадей;

$c = |\bar{c}|$;

$\bar{f} = -\frac{\bar{r}}{r} + \bar{v} \times \bar{c}$ - интеграл Лапласа;

$h = v^2 - \frac{2\mu}{r}$ - интеграл энергии;

$\alpha = -\frac{h}{\mu}$ - обратная большая полуось (со знаком);

$a = \frac{1}{\alpha}$ - большая полуось (со знаком);

$p = \frac{c^2}{\mu}$ - фокальный параметр;

$e = 1 + \frac{p}{\mu} h$ - эксцентриситет;

$n = \frac{\sqrt{\mu}}{a^{3/2}}$ - среднее движение;

$T = \frac{2\pi}{n}$ - период;

i - наклонение орбиты;

ω - аргумент перицентра;

Ω - долгота восходящего узла;

$f_2 = e\cos\omega$, $f_1 = e\sin\omega$ - проекции вектора $\frac{\bar{f}}{\mu}$ на линию узлов и перпендикулярное к ней направление в плоскости орбиты;

$r_\pi = \frac{p}{1 + e}$ - радиус перицентра

$r_\Omega = \frac{p}{1 + f_2}$ - радиус восходящего узла;

t_π, t_Ω - времена прохождения через перицентр и восходящий узел.

Наряду с $\bar{r}$, $\bar{v}$, t в работе используются следующие динамические параметры:

ϑ - истинная аномалия;

$u = \omega + \vartheta$ – аргумент широты;

$M_\pi = n(t-t_\pi)$, $M_\Omega = n(t-t_\Omega)$ – средние аномалии, отсчитываемые от перицентра и узла;

t_p – регуляризованное время, определяемое из дифференциального уравнения

$\frac{dt_p}{dt} = \frac{\sqrt{\mu}}{r}$, $t_p(t_o) = 0$; согласно [2],

$$t_p = \alpha\sqrt{\mu}(t-t_o) + \frac{\bar{r}^T\bar{v} - \bar{r}_o^T\bar{v}_o}{\sqrt{\mu}};$$

$\dot{r} = \frac{dr}{dt} = \frac{\bar{r}^T\bar{v}}{r}$ – радиальная скорость;

$$r' = \frac{dr}{dt_p} = \frac{\bar{r}^T\bar{v}}{\sqrt{\mu}};$$

$$r'' = \frac{d^2r}{dt_p^2} = -\alpha r + 1.$$

2. УНИВЕРСАЛЬНЫЕ АЛГОРИТМЫ РАСЧЕТА ТЕКУЩИХ ФАЗОВЫХ КООРДИНАТ И МАТРИЦ ИЗОХРОННЫХ ПРОИЗВОДНЫХ ПЕРВОГО И ВТОРОГО ПОРЯДКА

2.1. Вычисление текущих фазовых координат (подпрограмма KEPLER)

По заданным векторам $\bar{r}_0$, $\bar{v}_0$ координат и скоростей в момент t_0 и интервалу $t - t_0$ требуется вычислить векторы $\bar{r}$, $\bar{v}$ в момент t. Для этой цели используется следующий универсальный алгоритм [I].

Вначале определяется регуляризованное время t_p. Оно находится из универсального уравнения Кеплера:

$$(t-t_0) = r_0' \, t_p^2 \, C(y) + (1-r_0\alpha)\, t_p^3 S(y) + r_0 t_p, \qquad (2.1)$$

где r_0' находится согласно п. I.2, $y = \alpha t_p^2$, а функции $C(y)$ и $S(y)$ определяются формулами:

$$C(y) = \sum_{n=0}^{\infty} \frac{(-1)^n}{(2n+2)!}\, y^n = \begin{cases} \frac{1}{y}\,(1-\cos\sqrt{y}), & y>0, \\ \frac{1}{y}\,(1-\operatorname{ch}\sqrt{-y}), & y<0, \end{cases} \qquad (2.2)$$

$$S(y) = \sum_{n=0}^{\infty} \frac{(-1)^n}{(2n+3)!}\, y^n = \begin{cases} \frac{1}{y\sqrt{y}}\,(\sqrt{y} - \sin\sqrt{y}), & y>0, \\ \frac{1}{y\sqrt{-y}}\,(\sqrt{-y} - \operatorname{sh}\sqrt{-y}), & y<0. \end{cases} \qquad (2.3)$$

В подпрограмме KEPLER функции $C(y)$, $S(y)$ вычисляются рядами при $|y| \leq 0{,}1$.

Для нахождения t_p из уравнения (2.I) используется итерационный метод Ньютона, в котором приближенное решение на $(n+1)$-ой итерации выражается через предыдущее приближенное решение по формуле:

$$t_p^{(n+1)} = t_p^{(n)} - \frac{\sqrt{\mu}(t^{(n)}-t_0) - \sqrt{\mu}(t-t_0)}{r_{t=t^{(n)}}}, \qquad (2.4)$$

где $\sqrt{\mu}(t^{(n)}-t_0)$ правая часть (2.I) при $t = t^{(n)}$,

а функция $r = r_0' \left[t_p - \alpha t_p^3 S(y)\right] + (1-r_0\alpha)\, t_p^2\, C(y) + r_0$ (2.5)

определяет радиус в зависимости от t_p. Итерации начинаются с $t_p^{(0)} = \sqrt{\mu}(t-t_0)/\, r_0$.

Если итерационный процесс (2.4) плохо сходится, то в качестве нулевого приближения берется решение уравнения (2.I) для интервала $(t-t_0)/2$.

При расчете фазовых координат для эллиптического движения используется периодичность траектории и t_p ищется для интервала, который является остатком от деления $t-t_0$ на период обращения.

После нахождения t_p векторы координат и скоростей вычисляются по формуле:

$$\bar{r} = \left[1 - \frac{t_p^2}{r_0} C(y)\right] \bar{r}_0 + \left[t - t_0 - \frac{t_p^3}{\sqrt{\mu}} S(y)\right] \bar{v}_0 ,$$

$$\bar{v} = \frac{\sqrt{\mu}\, t_p}{r r_0} \left[y S(y) - 1\right] \bar{r}_0 + \left[1 - \frac{t_p^2}{r} C(y)\right] \bar{v}_0 . \qquad (2.6)$$

2.2. Вычисление матрицы первых изохронных производных (подпрограмма DERIV1)

По заданным $\mu, t-t_o, \bar{x}, \bar{x}_o$ требуется вычислять матрицу изохронных производных

$$\Phi = \frac{d\bar{x}}{d\bar{x}_o} = \left[\frac{dx_i}{dx_{oj}}\right]. \qquad (2.7)$$

Для этого используется предложенный в [2] алгоритм, пригодный для любых типов орбит в задаче двух тел. Этот алгоритм сводится к решению линейной алгебраической системы уравнений:

$$A\Phi = A_o,$$

где $A = A(t)$, $A_o = A(t_o)$ – матрицы 6-го порядка. Первые пять строк матрицы A находятся из формул:

$$\left.\begin{aligned} A_i &= [\bar{w}_i^T, \bar{u}_i^T], \\ A_{i+2} &= [(\nu\bar{r}\times\bar{u}_i + \bar{v}\times\bar{w}_i)^T, (\bar{p}_i\times\bar{o} - \bar{r}\times\bar{w}_i)^T], \end{aligned}\right\} i = 1,2 \qquad (2.8)$$

$$A_5 = [\nu\bar{r}^T, \bar{v}^T],$$

где

$$\nu = \frac{\mu}{r^3}, \qquad (2.9)$$

$$\bar{u}_i = \bar{p}_i\times\bar{r}, \quad \bar{w}_i = \bar{v}\times\bar{p}_i, \quad i = 1,2,$$

а $\bar{p}_1, \bar{p}_2$ – векторы, выбираемые из условия независимости строк $A_1 - A_6$ следующим образом: для орбит, близких к прямолинейным и параболическим, $\bar{p}_1, \bar{p}_2$ – орты системы координат, соответствующие наименьшим по модулю компонентам вектора $\bar{r}_o$; в остальных случаях – $\bar{p}_1 = \bar{r}/r$, $\bar{p}_2 = \bar{v}/v$.

Шестая строка матрицы A при $h\alpha > \varepsilon/r_o$, где $\varepsilon > 0$ – заданная малая величина *) (движение отлично от параболического), находится по формуле

$$A_6 = [-3\nu(t-t_o)\bar{r}^T + \bar{v}^T, 2\bar{r}^T - 3(t-t_o)\bar{v}^T]. \qquad (2.10)$$

---

*) В подпрограмме DERIV1 принимается ε=0,01.

При $|\alpha|\leqslant \varepsilon/r_o$ (параболическое и близкое к нему движение) A_6 вычисляется по формуле

$$A_6 = \left[(\mu\dot{r}-2\alpha S)\,\bar{r}^T + \dot{S}\bar{V}^T,\quad \dot{S}\bar{r}^T - 2S\bar{V}^T \right], \qquad (2.11)$$

где использованы обозначения

$$\dot{S} = c^2 - \mu r,$$

$$S = c^2(t-t_o) - \mu R_1, \qquad (2.I2)$$

$$R_1 = \int_{t_o}^{t} r\,dt = \frac{1}{\sqrt{\mu}}\left(r^2 t_p - rr' t_p^2 + \frac{rr''+r'^2}{3} t_p^3 + S_1\right),$$

$$S_1 = 2t_p^4 \sum_{n=2}^{\infty} \left[\frac{2^{2n-2}(r''^2-\alpha r'^2)-r''}{2n+1}\, t_p - (2^{2n-2}r''-1)r' \right] \frac{(-\alpha t_p^2)^{n-2}}{(2n)!},$$

а c, L, t_p, r, r', r'' определены в п. I.2. При $\alpha \neq 0$ интеграл R_1 вычисляется по конечной формуле

$$R_1 = \frac{3\sqrt{\mu}(t-t_o) - pt_p - rr' + r_o r_o'}{2\alpha\sqrt{\mu}}.$$

2.3. Вычисление матриц вторых изохронных производных (подпрограмма DERIV2)

По заданным $\mu, t-t_o, \bar{x}, \bar{x}_o$ требуется найти матрицы изохронных производных второго порядка

$$\Psi^k = \left[\Psi^k_{ij}\right] = \left[\frac{d^2 x_k}{dx_{oi}\,dx_{oj}}\right], \quad k = 1,\ldots,6.$$

Согласно [2] эти матрицы находятся из формул

$$\Psi^k = \sum_{l=1}^{6} a_{kl}\,(D_o^l - \Phi^T D^l \Phi), \quad k = 1,\ldots,6,$$

где a_{kl} – элементы матрицы A^{-1} (матрица A определяется из (2.8)–(2.II)), Φ – матрица изохронных производных первого порядка (алгоритм вычисления Φ приведен в п. 2.2), $D^l = D^l(t)$, $D_o^l = D^l(t_o)$, $l=1,\ldots,6$ –

— матрицы 6-го порядка. Для нахождения матрицы D^l, D^l_o введем обозначения:

$$\left.\begin{aligned} P_l &= \begin{bmatrix} 0 & p_{lz} & -p_{ly} \\ -p_{lz} & 0 & p_{lx} \\ p_{ly} & -p_{lx} & 0 \end{bmatrix}, \\ Q_l &= \bar{v}^T \bar{p}_l I + \bar{v}\bar{p}_l^T - 2\,\bar{p}_l \bar{v}^T, \\ R_l &= \bar{r}\bar{p}_l^T + \bar{p}_l \bar{r}^T, \end{aligned}\right\} \quad l = 1,2$$

$$G = \nu\left(3\frac{\bar{r}\bar{r}^T}{r^2} - I\right), \quad (I - \text{единичная матрица}),$$

где ν определяется из (2.9), $\bar{p}_1, \bar{p}_2$ — векторы, определенные в п. 2.2. Первые пять матриц D^l находятся из формул

$$\left.\begin{aligned} D^l &= \begin{bmatrix} 0 & P_l \\ P_l^T & 0 \end{bmatrix}, \\ D^{l+2} &= \begin{bmatrix} -\nu R_l + \bar{r}\bar{p}_l G & Q_l \\ Q_l & R_l - 2\bar{r}^T \bar{p}_l I \end{bmatrix}, \end{aligned}\right\} \quad l = 1,2$$

$$D^5 = \begin{bmatrix} -G & 0 \\ 0 & I \end{bmatrix}.$$

Матрица D^6 при $|\alpha| > \varepsilon/r_o$, где ε определяется так же, как и в п. 2.2 (движение отлично от параболического), находится по формуле

$$D^6 = \begin{bmatrix} 3(t-t_o)G & I \\ 2I & -3(t-t_o)I \end{bmatrix}.$$

При $|\alpha| \leqslant \varepsilon/r_o$ (параболическое и близкое к нему движение) D^6 вычисляется по формуле

$$D^6 = \begin{bmatrix} 2sG + \mu\dot{r}I + \bar{v}\bar{f}^T & \dot{s}I - \bar{r}\bar{f}^T \\ \dot{s}I + \bar{r}\bar{f}^T & -2sI \end{bmatrix} - \begin{bmatrix} \nu\bar{r} \\ \bar{v} \end{bmatrix} \bar{\xi}^T,$$

где

$$\bar{\xi} = 3(t-t_o)\begin{bmatrix} \bar{v}\times\bar{c} \\ \bar{c}\times\bar{r} \end{bmatrix} - 6R_2 \begin{bmatrix} \nu\bar{r} \\ \bar{v} \end{bmatrix} + \begin{bmatrix} \bar{c}\times\bar{r} \\ 2r^2\bar{v} \end{bmatrix},$$

$$R_2 = \int_{t_o}^{t} r^2 dt = \frac{1}{\sqrt{\mu}}\left[r^2 t_p\left(r - \frac{3}{2} r' t_p\right) + r t_p^3 \left(\frac{r r'}{2} + r'^2\right) + \right.$$

$$\left. + \frac{r t_p^4}{8}\left(\frac{r'' t_p}{5} - r'\right)(7r''-1) + r'^2 t_p^4\left(\frac{5r''-2}{10} - \frac{r'}{4}\right) + S_2 \right].$$

$$\delta_2=3t_p^6\sum_{n=3}^{\infty}\left[\frac{qr''-(m(n)r''-2^{2n-1})\alpha r'^2}{2n+1}\,t_p-(q-m(n-1)\,\alpha\, r'^2)r'\right]\frac{(-\alpha t_p^2)^{n-3}}{(2n)!},$$

$$m(n) = \frac{3^{2n}-1}{4},$$

$$q = \left[3m(n-1)+1\right] r''^2-2^{2n-1}r''+1,$$

величины c, f, t_p, r, r', r'' определены в п.I.2, а $e, \dot{a}$ определяются из (2.I2). При $\alpha \neq 0$ интеграл R_2 вычисляется по конечной формуле

$$R_2 = \frac{5\sqrt{\mu}\,(t-t_o)-3pt_p+(1-2\alpha r)rr'-(1-2\alpha r_o)r_o r_o'-\frac{4}{3}(r'^3-r_o'^3)}{2\alpha^2\sqrt{\mu}}.$$

2.4. <u>Вычисление производных по гравитационному параметру (подпрограмма DERM1, DERM2)</u>

По заданным $\mu, t-t_o, \bar{x}, \bar{x}_o$ требуется вычислить производные

$$\frac{d\bar{x}}{d\mu};\ \frac{d^2\bar{x}}{d\mu\, d\bar{x}_o};\ \frac{d^2\bar{x}}{d\mu^2}.$$

Согласно [2] эти производные находятся из формул

$$\frac{d\bar{x}}{d\mu} = \frac{1}{3\mu}(\bar{x}-\Phi\bar{x}_o),$$

$$\frac{d^2\bar{x}}{d\mu\, d\bar{x}_o} = \frac{d\Phi}{d\mu} = -\frac{1}{3\mu}\begin{bmatrix}\bar{x}_o^T\Psi^1\\ \cdots\\ \bar{x}_o^T\Psi^6\end{bmatrix},$$

$$\frac{d^2\bar{x}}{d\mu^2} = -\frac{1}{3\mu}\left(2\frac{d\bar{x}}{d\mu}+\frac{d\Phi}{d\mu}\bar{x}_o\right).$$

[illegible] Φ, Ψ^k, $k=1,\dots,6$ определены в пп. 2.2, 2.3.

3. АЛГОРИТМЫ ВЗАИМНОГО ПЕРЕХОДА МЕЖДУ ПРЯМОУГОЛЬНЫМИ ФАЗОВЫМИ КООРДИНАТАМИ И ЭЛЕМЕНТАМИ ОРБИТЫ

3.1. Пояснительные замечания к алгоритмам

При описании движения в задаче двух тел обычно используются различные совокупности элементов орбиты. Приведенные ниже алгоритмы разработаны для некоторых распространенных совокупностей. Наиболее универсальной из них является следующая совокупность элементов*), пригодная для любых непрямолинейных кеплеровских орбит:

$$\bar{y}_0 = (p,\ f_1,\ f_2,\ i, \Omega,\ u). \qquad (3.1)$$

Вектор $\bar{y}_0$ будем называть основной совокупностью элементов орбиты. Другие совокупности получаются заменой следующих элементов:

p – на a, h или T;

f_1, f_2 – на e, ω;

i – на $\cos i$;

u – на $t-t_\pi$, $t-t_\Omega$, M_π или M_Ω.

Для рассматриваемых угловых величин приняты следующие диапазоны:

$$0 \leqslant \omega, \Omega, u,\ M_\pi\ ,\ M_\Omega < 2\pi,\ 0 \leqslant i \leqslant \pi\ . \qquad (3.2)$$

Для элиптических орбит принято

$$0 \leqslant t-t_\pi\ ,\ t-t_\Omega < T. \qquad (3.3)$$

*) Под элементами орбиты мы будем понимать также динамические параметры движения на заданный момент времени t.

В подпрограммах, описанных в Приложении, выбором нужной со-совокупности элементов орбиты управляет трехзначное число

$$N_{эл} = \nu_1 \nu_2 \nu_3 \qquad (3.4)$$

(здесь ν_1, ν_2, ν_3 – десятичные цифры). Соответствие числа $N_{эл}$ и совокупности элементов орбиты приведено в табл.I.

Таблица I

Соответствие управляющего числа $N_{эл} = \nu_1 \nu_2 \nu_3$ и используемых элементов орбиты

ν_k	k=1	k=2	k=3
0	p	f_1, f_2, i	u
1	a	e, ω, i	$t-t_\pi$
2	h	$f_1, f_2, \cos i$	$t-t_\Omega$
3	T	$e, \omega, \cos i$	M_π
4	–	–	M_Ω

Например, числу $N_{эл}$=232 соответствует совокупность

$\vec{y} = (h, e, \omega, \cos i, \Omega, t-t_\Omega)$.

Всего таким образом можно получить 80 различных совокупностей элементов орбиты.

3.2. Вычисление времени перелета между двумя положениями на орбите (подпрограмма-функция DTIME)

Требуется определить время перелета t_2-t_1 (со знаком) между двумя положениями на орбите, заданными радиусами r_1, r_2, истинной аномалией ϑ_1 первого положения и угловой дальностью перелета φ. При этом знак интервала времени t_2-t_1 определяется знаком φ.

Воспользуемся универсальным алгоритмом, предложенным в [I]. Запишем угол φ в виде

$$\varphi = \varphi_0 + 2\pi k, \quad -2\pi < \varphi_0 < 2\pi, \; k=0, \pm 1, \pm 2, \ldots$$

(для неэллиптических орбит $k=0$). Тогда

$$t_2-t_1 = \frac{\operatorname{sign}\varphi}{\sqrt{\mu}}\left[\left(\frac{g}{C(x)}\right)^{\frac{3}{2}} S(x) - w\left(\frac{g-d}{C(y)}\right)^{\frac{3}{2}} S(y)\right] + kT,$$

где $C(x)$, $S(x)$ – функции, определяемые из (2.2), (2.3),

$$d = \sqrt{r_1^2 + r_2^2 - 2r_1 r_2 \cos\varphi},$$

$$g = \frac{r_1 + r_2 + d}{2},$$

$$w = \operatorname{sign} \sin|\varphi|,$$

$$x = \gamma^2 \operatorname{sign}\alpha, \quad y = \delta^2 \operatorname{sign}\alpha,$$

$$\gamma = \begin{cases} \pi + w_1\left(2 \arcsin\sqrt{\dfrac{\alpha g}{2}} - \pi\right), & \alpha \geq 0, \\ 2 \operatorname{arsh}\sqrt{\dfrac{-\alpha g}{2}}, & \alpha < 0, \end{cases}$$

$$w_1 = \begin{cases} w, & \sin\vartheta_1 \sin(\vartheta_1+\varphi) \geq 0, \\ -\operatorname{sign}(\varphi \sin\vartheta_1), & \sin\vartheta_1 \sin(\vartheta_1+\varphi) < 0, \end{cases}$$

$$\delta = \begin{cases} 2 \arcsin \sqrt{\frac{d(g-d)}{2}}, & d \geq 0, \\ 2 \operatorname{arsh} \sqrt{-\frac{d(g-d)}{2}}, & d < 0 \end{cases}$$

(при вычислениях удобно воспользоваться формулой
$\operatorname{arsh} z = \ln(z + \sqrt{1+z^2})$).

3.3. Расчет элементов орбиты по прямоугольным фазовым координатам (подпрограмма ABEL)

По заданным μ, $\bar{x}$ и числу $N_{эл}$ требуется вычислять определяемую этим числом совокупность элементов орбиты $\bar{y}$.

Основная совокупность элементов орбиты (3.I) вычисляется по следующим формулам:

$$p = \frac{c^2}{\mu},$$
$$\cos i = \frac{c_z}{c}, \quad \sin i \geq 0;$$

$$\cos\Omega = -\frac{c_y}{c \sin i}, \quad \sin\Omega = \frac{c_x}{c \sin i}.$$

При $i=0$ принимается $\Omega = 0$.

$$\cos u = \frac{1}{r}\left[(-r_x \sin\Omega + r_y \cos\Omega)c_z + r_z \sin i\right],$$
$$\sin u = \frac{1}{r}(r_x \cos\Omega + r_y \sin\Omega);$$

$$f_1 = \left(\frac{p}{r} - 1\right)\sin u - \frac{p}{r}\dot{r}\cos u;$$

$$f_2 = \left(\frac{p}{r} - 1\right)\cos u + \frac{p}{r}\dot{r}\sin u.$$

Остальные элементы орбиты, определенные в п.3.I, вычисляются через основные элементы следующим образом:

$$e = \sqrt{f_1^2 + f_2^2}$$

$$\sin\omega = \frac{f_1}{e}, \quad \cos\omega = \frac{f_2}{e}.$$

При $e = 0$ принимается $\omega = 0$.

$$a = \frac{p}{1-e^2}, \quad h = -\frac{\mu}{p}(1-e^2), \quad T = 2\pi\frac{a^{3/2}}{\sqrt{\mu}}.$$

Для вычисления интервалов времени $t-t_\pi$ или $t-t_\Omega$ используется алгоритм, приведенный в п.3.I. При этом необходимые исходные величины α, r_π, r_Ω определены в п.I.2,

$$\vartheta_1 = \begin{cases} 0, & \text{если вычисляется } t-t_\pi, \\ -\omega, & \text{если вычисляется } t-t_\Omega, \end{cases}$$

$$\varphi = u - \omega - \vartheta_1 + 2\pi k, \quad k = 0, \pm 1,$$

где, при соблюдении условия (3.2), число k выбирается на эллиптической орбите из условия (3.3), а на неэллиптической орбите – таким образом, чтобы получить правильный знак при $t-t_\pi$ и $t-t_\Omega$.

Отметим, что на неэллиптической орбите t_Ω может не существовать. В этом случае в подпрограмме ABEL вместо $t-t_\Omega$ вычисляется $t-t_\pi$ и печатаются соответствующие пояснения.

Средние аномалии M_π, M_Ω вычисляются по приведенным в п.I.2 формулам.

3.4. Расчет прямоугольных фазовых координат по элементам орбиты (подпрограмма ELAB)

Пусть заданы μ, число $N_{эл}$ и определяемая этим числом совокупность элементов орбиты $\bar{y}$. Требуется вычислить вектор текущих фазовых координат $\bar{x}$.

Будем рассматривать 2 случая.

а) В совокупность $\bar{y}$ входит аргумент широты u. В этом случае векторы $\bar{r}, \bar{v}$ вычисляются по формулам

$$\bar{r} = r\bar{r}^{\circ}, \quad \bar{v} = v_r\bar{r}^{\circ} + v_n\bar{n}^{\circ},$$

$$r = \frac{p}{1 + f_1 \sin u + f_2 \cos u},$$

$$v_r = \sqrt{\frac{\mu}{p}}(f_2 \sin u - f_1 \cos u), \quad v_n = \frac{\sqrt{\mu p}}{r},$$

$$\bar{r}^{\circ} = \begin{bmatrix} \cos\Omega \cos u - \sin\Omega \sin u \cos i \\ \sin\Omega \cos u + \cos\Omega \sin u \cos i \\ \sin u \sin i \end{bmatrix}, \qquad (3.5)$$

$$\bar{n}^{\circ} = \begin{bmatrix} -\cos\Omega \sin u - \sin\Omega \cos u \cos i \\ -\sin\Omega \sin u + \cos\Omega \cos u \cos i \\ \cos u \cos i \end{bmatrix}.$$

Если входящие в (3.5) элементы p, f_1, f_2 не используются в совокупности $\bar{y}$, то они вычисляются по используемым в $\bar{y}$ элементам с помощью формул, приведенных в п.I.2.

б) В совокупность $\bar{y}$ входит параметр $t-t_0$ или $n(t-t_0)$, где $t_0 = t_\pi$ или $t_0 = t_\Omega$. В этом случае сначала вычисляется вектор $\bar{x}_c = \bar{x}(t_0)$, для чего в (3.5) следует принять

$$u = \begin{cases} \omega, & \text{если } t_0 = t_\pi; \\ 0, & \text{если } t_0 = t_\Omega. \end{cases}$$

При известных $t-t_0$ и $\bar{x}_0$ вектор $\bar{x} = \bar{x}(t)$ определяется при помощи алгоритма, описанного в п.2.I.

3.5. Расчет совокупности элементов орбиты по другой совокупности элементов (подпрограмма ELEL)

Пусть заданы μ, число $N_{элI}$, определяемая этим числом совокупность элементов $\bar{y}_1$ и число $N_{эл2}$. Требуется вычислить совокупность элементов $\bar{y}_2$, определяемую числом $N_{эл2}$.

В подпрограмме ELEL использован следующий способ вычисления $\bar{y}_2$: сначала по μ, $N_{элI}$, $\bar{y}_1$ при помощи алгоритма, приведенного в п.3.4, вычисляется вектор $\bar{x}$; затем по μ, $\bar{x}$, $N_{эл2}$ с помощью алгоритма, описанного в п.3.3, вычисляется $\bar{y}_2$.

3.6. Вычисление производных от основной совокупности элементов орбиты по другим элементам орбиты (подпрограмма PELOEL)

Пусть заданы μ, основная совокупность элементов $\bar{y}_0$, определяемая из (3.I), число $N_{эл}$ и определяемая этим числом совокупность $\bar{y}$. Требуется вычислить матрицу производных

$$P_{эл} = \frac{d\bar{y}_0}{d\bar{y}} .$$

В зависимости от того, какие элементы орбиты входят в совокупность $\bar{y}$, ненулевые элементы матрицы $P_{эл}$ определяются из следующих соотношений

$$\frac{dp}{da} = \frac{p}{a}, \quad \frac{dp}{dh} = -\frac{p}{h}, \quad \frac{dp}{dT} = \frac{2}{3}\frac{p}{T},$$

$$\frac{dp}{de} = -2ae, \quad \frac{dp}{df_i} = -2af_i, \quad i=1,2,$$

$$\frac{df_1}{de} = \sin\omega, \quad \frac{df_2}{de} = \cos\omega, \quad \frac{df_1}{d\omega} = f_2, \quad \frac{df_2}{d\omega} = -f_1,$$

$$\frac{di}{d\cos i} = \frac{1}{\sin i} .$$

Если в совокупность $\bar{y}$ вместо u используется $t-t_0$, где $t_0=t_{\pi}$ или $t_0=t_{\Omega}$, то

$$\frac{du}{dp} = -\frac{3c}{2pr^2}(t-t_0), \quad \frac{du}{da} = -\frac{3c}{2ar^2}(t-t_0),$$

$$\frac{du}{dh} = \frac{3c}{2hr^2}(t-t_0), \quad \frac{du}{dT} = -\frac{c}{Tr^2}(t-t_0),$$

$$\frac{du}{d(t-t_0)} = \frac{c}{r^2};$$

здесь $c = \sqrt{\mu p}$, $r=p/(1+f_1 \sin u + f_2 \cos u)$.

Если же в $\bar{y}$ используется M_{π} или M_{Ω}, то

$$\frac{du}{dp} = \frac{du}{da} = \frac{du}{dh} = \frac{du}{dT} = 0,$$

$$\frac{du}{dM_{\pi}} = \frac{du}{dM_{\Omega}} = \left(\frac{a}{r}\right)^2 \sqrt{1-e^2}.$$

Приведем формулу для производной $\frac{du}{de}$, вид которой зависит от того, какой из элементов p, a, h, T, а также какой из параметров $t-t_0$, $n(t-t_0)$ входят в $\bar{y}$ (здесь, как и ранее, $t_0=t_{\pi}$ или $t_0=t_{\Omega}$):

$$\frac{du}{de} = \frac{a}{p}\left(1+\frac{p}{r}\right)\sin\vartheta - U_1 + U_2.$$

Здесь $\vartheta = u-\omega$,

$$U_1 = \begin{cases} 3\dfrac{ce}{r^2(1-e^2)}(t-t_0), & \text{если } p \in \bar{y},\ M_{\pi},\ M_{\Omega} \notin \bar{y}; \\ 0, & \text{если } p \notin \bar{y} \text{ или } M_{\pi} \text{ или } M_{\Omega} \in \bar{y}; \end{cases}$$

$$U_2 = \begin{cases} \left(\dfrac{r_{\Omega}}{r}\right)^2 \dfrac{a}{p}(2+f_2)\sin\omega, & \text{если } t_0=t_{\Omega}; \\ 0, & \text{если } t_0=t_{\pi}. \end{cases}$$

Величина r_{Ω} определена в п.I.2.

Производная $\frac{du}{d\omega}$ также зависит от входящих в $\bar{y}$ параметров:

$$\frac{du}{d\omega} = \begin{cases} 1, & \text{если } t_o = t_\pi ; \\ \frac{r_\Omega}{p}\left(1+\frac{r_\Omega}{r}\right)(\cos\omega - \cos\vartheta)e, & \text{если } t_o = t_\Omega . \end{cases}$$

Производные $\frac{du}{df_i}$, $i=1,2$, вычисляются по формулам

$$\frac{du}{df_1} = S_1 \sin\omega + S_2 \cos\omega ,$$

$$\frac{du}{df_2} = S_1 \cos\omega - S_2 \sin\omega ,$$

где $S_1 = \frac{du}{de}$, $S_2 = \frac{1}{e}\frac{du}{d\omega}$.

3.7. Вычисление производных от одних элементов орбиты по другим элементам (подпрограмма PELEL)

Пусть заданы μ , числа $N_{элI}$, $N_{эл2}$ и определяемые этими числами совокупности элементов орбиты $\bar{y}_1$, $\bar{y}_2$. Требуется найти матрицу производных

$$P_{эл} = \frac{d\bar{y}_1}{d\bar{y}_2} .$$

С учетом приведенных выше алгоритмов эта матрица вычисляется по формуле

$$P_{эл} = P_{элI}^{-1} P_{эл2} ,$$

где матрицы $P_{элi} = \frac{d\bar{y}_o}{d\bar{y}_i}$, $i=1,2$, находятся по алгоритму, описанному в п.3.6, а основная совокупность элементов $\bar{y}_o$ вычисляется по совокупности $\bar{y}_1$ с помощью алгоритма, приведенного в п.3.5.

3.8. Расчет производных от прямоугольных фазовых координат по элементам орбиты (подпрограмма PABEL)

Пусть заданы μ, вектор $\bar{x}$, число $N_{эл}$ и совокупность элементов орбиты $\bar{y}$, определяемая этим числом. Требуется вычислить матрицу производных

$$P = \frac{d\bar{x}}{d\bar{y}}. \tag{3.7}$$

С этой целью сначала вычисляется матрица

$$P_0 = \frac{d\bar{x}}{d\bar{y}_0},$$

компоненты которой приведены в табл.2. Входящие в P_0 элементы из основной совокупности $\bar{y}_0$ рассчитываются по вектору $\bar{x}$ согласно приведенному в п.3.2 алгоритму. Затем с помощью алгоритма, описанного в п.3.5, вычисляется матрица

$$P = \frac{d\bar{y}_0}{d\bar{y}}.$$

Матрица (3.7) вычисляется по формуле $P = P_0 P_{эл}$.

3.9. Расчет производных от элементов орбиты по прямоугольным фазовым координатам (подпрограмма PELAB)

По заданным μ, $N_{эл}$, $\bar{y}$, $\bar{x}$ требуется вычислить матрицу производных

$$P' = \frac{d\bar{y}}{d\bar{x}}.$$

В подпрограмме PELAB для этой цели использована формула

$$P' = P^{-1},$$

где $P = \dfrac{d\bar{x}}{d\bar{y}}$ – матрица, рассчитанная по описанному в п.3.8 алгоритму.

Таблица 2

МАТРИЦА ПРОИЗВОДНЫХ $P_o = d\bar{x}/d\bar{y}_o$

	$/dp$	$/df_1$	$/df_2$	$/di$	$/d\Omega$	$/du$
$d\bar{r}/$	$\frac{\bar{r}}{p}$	$-\frac{r\bar{r}}{p}\sin u$	$-\frac{r\bar{r}}{p}\cos u$	$\frac{\bar{c}}{c}r\sin u$	$\begin{matrix}-r_y\\ r_x\\ 0\end{matrix}$	$\frac{r^2}{c}\bar{v}$
$d\bar{v}/$	$-\frac{\bar{v}}{2p}$	$-\frac{c}{p}\begin{bmatrix}\cos\Omega\\ \sin\Omega\\ 0\end{bmatrix}$	$\frac{c}{p}\begin{bmatrix}-\sin\Omega\cos i\\ \cos\Omega\cos i\\ \sin i\end{bmatrix}$	$\frac{\bar{c}}{p}(f_2+\cos u)$	$\begin{matrix}-v_y\\ v_x\\ 0\end{matrix}$	$-\frac{c}{pr}\bar{r}$

3.10. Вычисление матрицы изохронных производных (второй способ) (подпрограмма DERIVE).

Приведенные выше алгоритмы позволяют по заданным $\mu, \bar{x}, \bar{x}_0, t-t_0$ вычислить матрицу

$$\Phi = \frac{d\bar{x}}{d\bar{x}_0} \qquad (3.8)$$

способом, отличным от описанного в п. 2.2. Для этого по $\bar{x}, t-t_0, \mu$ с помощью алгоритма, приведенного в п.3.3, рассчитываются совокупности элементов орбиты

$\bar{y} = (p, f_1, f_2, i, \Omega, t-t_\pi)$,

$\tilde{y}_i = y_i$, $i=1,\ldots,5$, $\tilde{y}_6 = y_6-(t-t_0)$

(при этом $K_{эл}=001$), а затем по алгоритму, описанному в пп.3.8, 3.9, вычисляются матрицы

$F = \frac{d\bar{x}}{d\bar{y}}$, $F_0 = \frac{d\tilde{y}}{d\bar{x}_0}$.

Матрица (3.8) рассчитывается по формуле

$\Phi = FF_0$

(здесь использовано равенство $\frac{d\bar{y}}{d\tilde{y}} = I$).

Следует отметить, что этот алгоритм менее универсален, чем приведенный в п.2.2 (не пригоден для круговых и прямолинейных орбит).

ПРИЛОЖЕНИЕ

ОПИСАНИЕ И ТЕКСТЫ ПОДПРОГРАММ НА ЯЗЫКЕ ФОРТРАН

Приводимые ниже описания подпрограмм составлены по следующему образцу:

назначение подпрограммы;

исходные данные и результат (в скобках в той же последовательности приведены соответствующие параметры движения, согласно введенным в пп.I.2, 3.I обозначениям);

примечание (может отсутствовать);

используемые подпрограммы и функции (исключая библиотечные);

фортранный текст.

Кроме того, приведены тексты вспомогательных подпрограмм и функций, например, подпрограмм линейной алгебры.

При составлении подпрограмм приняты меры для предотвращения потери точности и переполнения.

ПОДПРОГРАММА KEPLER(GM,DT,XO,X)

Служит для вычисления текущих фазовых координат $\bar{x}$ по их начальным значениям $\bar{x}_0$.

Исходные данные: GM,DT,XO(6)(μ,$t-t_0$,$\bar{x}_0$).

Результат: x(6)($\bar{x}$).

Примечание. Если исходные данные соответствуют гиперболической орбите и выбранный масштаб этих данных приводит к недопустимо большим промежуточным числам, то печатается соответствующий текст и счет прекращается.

Используемые подпрограммы и функции: VMOD, SPR,CS.

```
      SUBROUTINE KEPLER(GM,DT,X0,X)
      DIMENSION X0(6),X(6),R0(3),V0(3)
      DATA PI2/6.28318530718/
     *     ,EPS/1.E-10/
C EPS-ТОЧНОСТЬ РЕШ.УРАВН.КЕПЛЕРА В ЕДИНИЦАХ SQRT(AR)
     *     ,EV/-600./
     *     ,EAL/1.E-6/
      DO 10 K=1,3
      R0(K)=X0(K)
   10 V0(K)=X0(K+3)
      C1=SQRT(GM)
      AR0=VMOD(R0,3)
      AL=2./AR0-VMOD(V0,3)**2/GM
      IF((AL*AR0).GT.EAL)
     *                      DT=AMOD(DT,PI2/C1/AL/SQRT(AL))
C УМЕНЬШЕНИЕ ВРЕМЕНИ НА ЦЕЛОЕ ЧИСЛО ПЕРИОДОВ
      T0=0.
C T0-ВРЕМЯ,НА КОТОРОЕ ЗАДАНО X0( ЕСЛИ СХОДИМОСТЬ ПЛОХАЯ,ТО
C ВЫЧИСЛЯЮТСЯ НОВЫЕ X0(ПЕРЕХОД НА МЕТКУ 103) И T0(СМ.
C ФОРТР. ТЕКСТ ПОСЛЕ МЕТКИ 2)
  101 C2=SPR(R0,V0,3)/C1
      C3=1.-AR0*AL
      C4=C1/AR0
      T=DT-T0
      N=0
  100 TA=C1*T/AR0
C TA  - РЕГУЛЯРИЗОВАННОЕ ВРЕМЯ
      FA=1.E10
    1 F1=TA**2
      Y=AL*F1
      IF(Y.LT.EV)
     *                 GOTO 103
```

```
      F2=F1*TR
      CALL CS(V,C,S)
      F3=F1*C
      F4=F2*S
      F5=AL*F4-TR
      AR=AR0+C3*F3-C2*F5
      F60=F6
      F6=(C2*F3+C3*F4+AR0*TR-C1*T)/AR
      IF(ABS(F6).GT.ABS(F60))
     -                                GOTO 103
      TR=TR-F6
      ETR=EPS*SQRT(AR)
      IF(ABS(F6).GT.ETR)
     -                          GOTO 1
      F=1.-F3/AR0
      G=T-F4/C1
      FT=F5*C4/AR
      GT=1.-F3/AR
      DO 2 K=1,3
      X(K)=F*R0(K)+G*V0(K)
      X(K+3)=FT*R0(K)+GT*V0(K)
      R0(K)=X(K)
    2 V0(K)=X(K+3)
      IF(N.EQ.0)
     -                 GOTO 3
      AR0=AR
      T0=T0+T
C ИЗМЕНЕНИЕ НАЧАЛЬНОГО ВРЕМЕНИ
      GOTO 101
  103 T=T/2.
C ВЫЧИСЛ.НОВОЙ НАЧ.ТОЧКИ ПРИ ПЛОХОЙ СХОДИМОСТИ
      N=N+1
      IF(N.LT.10)
     -                  GOTO 100
      PRINT 102
  102 FORMAT(X,'KEPLER:ИЗМЕНИТЬ МАСШТАБ ИСХОДНЫХ ДАННЫХ'
     *' ИЛИ УМЕНЬШИТЬ EV1 СТОП')
      STOP
    3 RETURN
      END
```

ПОДПРОГРАММА-ФУНКЦИЯ RINT(GM,DT,RO,VO,R,V).

Вычисляет $R_1 = \int_{t_0}^{t} r dt$.

Исходные данные: GM,DT,RO(3),VO(3),V(3), ($\mu, t-t_0, \bar{r}_0, \bar{v}_0, \bar{r}, \bar{v}$).

Результат: RINT(R_1).

Используемые функции: VMOD,SPR.

```
      FUNCTION RINT(GM,DT,R0,V0,R,V)
      DIMENSION R0(3),V0(3),R(3),V(3)
      DATA EPS0/1.E-2/
      DATA EPS/1.E-10/
      SGM=SQRT(GM)
      R0M=VMOD(R0,3)
      RM=VMOD(R,3)
      AL=2./RM-SPR(V,V,3)/GM
      R01=SPR(R0,V0,3)/SGM
      R1=SPR(R,V,3)/SGM
      R2=1.-AL*RM
      TAU=AL*SGM*DT+R1-R01
      IF (ABS(AL).GT.EPS0/R0M)
     -                                GO TO 10
      S=(RM**2-RM*R1*TAU+(RM*R2+R1**2)*TAU**2/3.)*TAU
      AN=4.
      AM=4.
      C=R2**2-AL*R1**2
      V=TAU**4/12.
      PV=-AL*TAU**2
    1 DS=((AM*C-R2)*TAU/(AN+1.)-(AM*R2-1.)*R1)*V
      IF (ABS(DS).LE.EPS*ABS(S))
     -                                GO TO 2
      S=S+DS
      AN=AN+2.
      AM=AM+4.
      V=V*PV/((AN-1.)*AN)
      GO TO 1
    2 RINT=S/SGM
      RETURN
   10 P=(2.-AL*RM)*RM-R1**2
      RINT=(3.*SGM*DT-P*TAU-RM*R1*R0M*R01)/(2.*AL*SGM)
      RETURN
      END
```

ПОДПРОГРАММА-ФУНКЦИЯ R2INT(GM,DT,RO,VO,R,V)

Вычисляет $R_2 = \int_{t_0}^{t} r^2 dt$.

Исходные данные: GM,DT,RO(3),VO(3),R(3),V(3)(μ, $t-t_0$, $\vec{r}_0$, $\vec{v}_0$, $\vec{r}$, $\vec{v}$).

Результат: R2INT(R_2).

Используемые функции: VMOD,SPR.

```
      FUNCTION R2INT(GM,DT,R0,V0,R,V)
      DIMENSION R0(3),V0(3),R(3),V(3)
      DATA EPS0/1.E-2/
      DATA EPS/1.E-10/
      SGM=SQRT(GM)
      R0M=VMOD(R0,3)
      RM=VMOD(R,3)
      AL=2./RM-SPR(V,V,3)/GM
      R01=SPR(R0,V0,3)/SGM
      R1=SPR(R,V,3)/SGM
      R2=1.-AL*RM
      TAU=AL*SGM*DT+R1-R01
      R12=R1**2
      TAU3=TAU**3
      IF (ABS(AL).GT.EPS0/R0M)
     -                                 GO TO 10
      S=(RM**2*(RM-1.5*R1*TAU)+RM*TAU**2*(0.5*RM*R2+R12)+
     *  (0.125*RM*(0.2*R2*TAU-R1)*(7.*R2-1.)+R12*((0.5*R2-
     *  0.2)*TAU-0.25*R1))*TAU3)*TAU
      AN=6.
      AM=32.
      BM1=20.
      C=AL*R12
      V=TAU3**2/240.
      PV=-AL*TAU**2
    1 BM2=9.*BM1+2.
      Q=(3.*BM1+1.)*R2**2-AM*R2+1.
      DS=((Q*R2-(BM2*R2-AM)*C)*TAU/(AN+1.)-(Q-BM1*C)*R1)*V
      IF (ABS(DS).LE.EPS*ABS(S))
     -                                 GO TO 2
      S=S+DS
      AN=AN+2.
      AM=AM+4.
      BM1=BM2
      V=V*PV/((AN-1.)*AN)
      GO TO 1
    2 R2INT=S/SGM
      RETURN
   10 P=(2.-AL*RM)*RM-R12
      TAL=2.*AL
      R2INT=(3.*SGM*DT-3.*P*TAU+(1.-TAL*RM)*RM*R1-(1.-TAL*R0M)*
     *    R0M*R01-(R1-R01)*(R12+(R1+R01)*R01)/0.75)/(TAL*SGM)/AL
      RETURN
      END
```

ПОДПРОГРАММА DERIV1(GM,DT,X,XO,FI)

Служит для вычисления матрицы изохронных производных $\Phi = \dfrac{d\bar{x}}{d\bar{x}_0}$.

Исходные данные: GM,DT,X(6),XO(6)($\mu, t-t_0, \bar{x}, \bar{x}_0$).

Результат: FI(6,6) (FI(I,J)=Φ_{ij}).

Дополнительный результат:

COMMON/BLPCS/P1(3),P2(3),C(3),S1,S2,PARAB
($\bar{p}_1, \bar{p}_2, \bar{c}$; если движение близко к параболическому, логическая переменная PARAB=.TRUE. и S1 = $2\left[c^2(t-t_0)-\mu\int_{t_0}^{t} r dt\right]$; S2 не используется).

COMMON/BLA/A(6,6)   (A^{-1};   - см.п.2.2).

Используемые подпрограммы и функции: PCHOIC,VPB,VMOD, SPR, RINT,MATRA,MATIN1   (обращение матрицы), MUM.

```
      SUBROUTINE DERIV1(GM,DT,X,X0,FI)
      DIMENSION X0(6),X(6),R0(3),V0(3),R(3),V(3),FI(6,6),A0(6,6)
     *          ,IND(6,2)
      COMMON /BLA/A(6,6)
      COMMON /BLPCS/P1(3),P2(3),C(3),S1,S2,PARAB
      LOGICAL PARAB
      DATA EPS/1.E-2/
      DO 1 I=1,3
      R0(I)=X0(I)
      V0(I)=X0(I+3)
      R(I)=X(I)
    1 V(I)=X(I+3)
      CALL VPB(R0,V0,C)
      U=GM/VMOD(R0,3)
      H=SPR(V0,V0,3)-2.*U
      PARAB=.FALSE.
      IF (ABS(H).LE.EPS*U)
     *                         PARAB=.TRUE.
      CALL PCHOIC(R0,V0)
      IF (PARAB)
     *            S1=2.*(SPR(C,C,3)*DT-GM*RINT(GM,DT,R0,V0,R,V))
      CALL MATRA(GM,DT,R,V,A)
      CALL MATRA(GM,0.,R0,V0,A0)
      CALL MATIN1(A,6,6,6,0,IND,NER,DET)
      CALL MUM(A,A0,FI,6,6,6)
      RETURN
      END
```

ПОДПРОГРАММА DERIV2(GM,DT,X,XO,FI,PSI)

Служит для вычисления матриц изохронных производных первого и второго порядка

$$\Phi = \frac{d\bar{x}}{d\bar{x}_o}, \quad \Psi^k = \frac{d}{d\bar{x}_o}\left(\frac{dx_k}{d\bar{x}_o}\right)^T = \frac{d^2 x_k}{dx_{oi}dx_{oj}},$$

Исходные данные: GM,DT,X(6),XO(6) ($\mu, t-t_o, \bar{x}, \bar{x}_o$).

Результат: FI(6,6),PSI(6,6,6) (FI(I,J)=Φ_{ij},PSI(I,J,K)=Ψ^k_{ij}).

Дополнительный результат:

COMMON/BLPCS/P1(3),P2(3),C(3),S1,S2,PARAB

COMMON/BLA/A(6,6)

(то же, что в DERIV1; если PARAB=.TRUE.,S2=$6\int_{t_o}^{t} r^2 dt$).

Используемые подпрограммы и функции: DERIV1,R2INT,MATRD, TRM,MUM.

```
      SUBROUTINE DERIV2(GM,DT,X,X0,FI,PSI)
      DIMENSION X0(6),X(6),R0(3),V0(3),R(3),V(3),FI(6,6),
     *          PSI(6,6,6),FIT(6,6),D(6,6),D0(6,6)
      COMMON /BLA/A(6,6)
      COMMON /BLPCS/PCS(10),S2,PARAB
      LOGICAL PARAB
      DO 3 I=1,3
      R0(I)=X0(I)
      V0(I)=X0(I+3)
      R(I)=X(I)
    3 V(I)=X(I+3)
      CALL DERIV1(GM,DT,X,X0,FI)
      DO 1 I=1,216
    1 PSI(I)=0.
      IF (PARAB)
     *              S2=6.*R2INT(GM,DT,R0,V0,R,V)
      DO 2 L=1,6
      CALL MATRD(GM,DT,R,V,D,L)
      CALL TRM(FI,FIT,6,6)
      CALL MUM(D,FI,D0,6,6,6)
      CALL MUM(FIT,D0,D,6,6,6)
      CALL MATRD(GM,0.,R0,V0,D0,L)
      DO 2 K=1,6
      DO 2 I=1,6
      DO 2 J=1,6
    2 PSI(I,J,K)=PSI(I,J,K)+A(K,L)*(D0(I,J)-D(I,J))
      RETURN
      END
```

ПОДПРОГРАММА DERM1(GM,DT,X,XO,FI,XM)

Служит для вычисления производных $\Phi = \frac{d\bar{x}}{d\bar{x}_0}$, $\bar{x}_\mu = \frac{d\bar{x}}{d\mu}$.

Исходные данные: GM,DT,X(6),XO(6)(μ, $t-t_0$, $\bar{x}$, $\bar{x}_0$).

Результат: FI(6,6),XM(6) (Φ, $\bar{x}_\mu$).

Используемые подпрограммы: DERIV1,MUMV.

```
      SUBROUTINE DERM1(GM,DT,X,X0,FI,XM)
      DIMENSION X0(6),X(6),FI(6,6),XM(6)
      CALL DERIV1(GM,DT,X,X0,FI)
      CALL MUMV(FI,X0,XM,6)
      U=1./3./GM
      DO 1 I=1,6
    1 XM(I)=U*(X(I)-XM(I))
      RETURN
      END
```

ПОДПРОГРАММА DERM2(GM,DT,X,XO,FI,PSI,XM,XMM,XMX)

Служит для вычисления производных $\Phi = \frac{d\bar{x}}{d\bar{x}_o}$,

$$\Psi^k = \frac{d}{d\bar{x}_o}\left(\frac{dx_k}{d\bar{x}_o}\right)^T = \left[\frac{d^2 x_k}{dx_{oi}dx_{oj}}\right], \quad k=1,\dots,6, \qquad \bar{x}_\mu = \frac{d\bar{x}}{d\mu},$$

$$\bar{x}_{\mu\mu} = \frac{d^2\bar{x}}{d\mu^2}, \qquad \Phi_\mu = \frac{d^2\bar{x}}{d\bar{x}_o d\mu} = \frac{d\Phi}{d\mu}.$$

Исходные данные: GM,DT,X(6),XO(6)(μ, $t-t_o$, $\bar{x}$, $\dot{\bar{x}}_o$).

Результат: FI(6,6),PSI(6,6,6),XM(6),XMM(6),XMX(6,6) (Φ, $\Psi^1,\dots,\Psi^6$, $\bar{x}_\mu$, $\bar{x}_{\mu\mu}$, Φ_μ).

Используемые подпрограммы: DERM1,DERIV2,MUMV.

```
      SUBROUTINE DERM2(GM,DT,X,X0,FI,PSI,XM,XMM,XMX)
      DIMENSION X0(6),X(6),FI(6,6),PSI(6,6,6),XM(6),XMM(6),
     *          XMX(6,6)
      CALL DERM1(GM,DT,X,X0,FI,XM)
      CALL DERIV2(GM,DT,X,X0,FI,PSI)
      DO 2 K=1,6
      DO 2 J=1,6
      U=0.
      DO 1 I=1,6
    1 U=U+X0(I)*PSI(I,J,K)
    2 XMX(K,J)=U
      CALL MUMV(XMX,X0,XMM,6)
      U=-2./3.
      DO 3 I=1,6
    3 XMM(I)=U*XM(I)+XMM(I)
      RETURN
      END
```

ПОДПРОГРАММА-ФУНКЦИЯ DTIME(GM,AL,R1,R2,TE1,FI)

Вычисляет время перелета t_2-t_1 (со знаком) между двумя положениями на орбите, заданными радиусами r_1, r_2, истинной аномалией ϑ_1 первого положения и угловой дальностью перелета φ.

Исходные данные: GM,AL,R1,R2,TE1,FI($\mu, \alpha, r_1, r_2, \vartheta_1, \varphi$).

Результат: DTIME(t_2-t_1).

Примечание. Знак t_2-t_1 равен знаку φ. Если на эллиптической орбите $\varphi = \varphi_0 + 2\pi k$, $-2\pi < \varphi_0 < 2\pi$, $k = \pm 1, \pm 2, \ldots$, то к t_2-t_1 прибавляется величина kT.

Используемая подпрограмма: CS.

```
      FUNCTION DTIME(GM,AL,R1,R2,TE1,FI)
      DATA PI2/6.28318530718/
      ASH(X)=ALOG(X+SQRT(1.+X**2))
      C=SQRT(R1**2+R2**2-2.*R1*R2*COS(FI))
      S1=(R1+R2+C)/2.
      S2=S1-C
      ALM=ABS(AL)/2.
      A=SQRT(ALM*S1)
      B=SQRT(ALM*S2)
      STE=SIN(TE1)
      W=SIGN(1.,SIN(ABS(FI)))
      IF (AL) 2,2,1
    1 G=2.*ASIN(A)
      D=2.*ASIN(B)
      W1=W
      IF (STE*SIN(TE1+FI).LT.0.)
     -                                W1=-FI*STE
      IF (W1.LT.0.)
     -                  G=PI2-G
      GO TO 3
    2 G=2.*ASH(A)
      D=2.*ASH(B)
    3 X=SIGN(G**2,AL)
      Y=SIGN(D**2,AL)
      CALL CS(X,CX,SX)
      CALL CS(Y,CY,SY)
      DT=(S1/CX*SQRT(S1/CX)*SX-W*S2/CY*SQRT(S2/CY)*SY)*
     *   SIGN(1./SQRT(GM),FI)
      IF (AL*R1.LE.1.E-8)
     -                       GO TO 6
      DT=DT+SIGN(AINT(ABS(FI)/PI2),FI)*PI2/AL/SQRT(AL*GM)
    6 DTIME=DT
      RETURN
      END
```

ПОДПРОГРАММА ABEL(GM,X,NEL,EL)

Служит для вычисления совокупности элементов орбиты $\bar{y}$, определяемой числом $N_{эл}$ (см. п.3.I).

Исходные данные: GM,X(6),NEL (μ, $\bar{x}$, $N_{эл}$).

Результат: EL(6) ($\bar{y}$).

Дополнительный результат: COMMON /BLEL4/ P,EX,W,U (p, e, ω, u)

COMMON/BLRC8/AR,AC,C(3),CO(3) ($r, c, \bar{c}, \bar{c}/c$);

COMMON/BLSC4/SOM,COM,SU,CU ($\sin\Omega$, $\cos\Omega$, $\sin u$, $\cos u$).

Примечание. Если заказанные элементы не существуют на заданной орбите (например, период T - на гиперболе), печатается соответствующий текст и счет прекращается.

Используемые подпрограммы и функции: VMOD,SPR,VPB,NORM, ATAN3,NUMEL,DTIME,PDIAGN.

```
      SUBROUTINE ABEL(GM,X,NEL,EL)
      DIMENSION X(6),EL(6),R(3),V(3)
      DIMENSION N(3)
      EQUIVALENCE (N1,N(1)),(N2,N(2)),(N3,N(3))
      COMMON /BLRC8/AR,AC,C(3),C0(3)
      EQUIVALENCE (CI,C0(3))
      COMMON /BLSC4/SOM,COM,SU,CU
      COMMON /BLEL6/P,EX,W,U
      DATA PI2/6.28318530718/
      DO 1 I=1,3
      R(I)=X(I)
    1 V(I)=X(I+3)
      AR=VMOD(R,3)
      AV=VMOD(V,3)
      VR=SPR(R,V,3)/AR
      IF (ABS(ABS(VR)/AV-1.).GT.1.E-8)
     *                                   GO TO 2
      K=4
      GO TO 27
    2 CALL VPB(R,V,C)
      CALL NORM(C,C0,AC)
      P=AC**2/GM
      SI=SQRT(ABS((1.-CI)*(1.+CI)))
      IF (SI.GT.1.E-8)
     *                      GO TO 3
      SOM=0.
      COM=1.
      GO TO 4
    3 SOM=C0(1)/SI
      COM=-C0(2)/SI
    4 SU=((-R(1)*SOM+R(2)*COM)*CI+R(3)*SI)/AR
      CU=(R(1)*COM+R(2)*SOM)/AR
      U=ATAN3(SU,CU)
      ECT=P/AR-1.
      EST=P*VR/AC
      F1=ECT*SU-EST*CU
      F2=ECT*CU+EST*SU
      EX=SQRT(F1**2+F2**2)
      EX1=(1.-EX)*(1.+EX)
      CALL NUMEL(NEL,N)
      GO TO (6,5,6,7),N1
    5 IF (ABS(EX1).GT.1.E-8)
     *                            GO TO 6
      K=2
      GO TO 27
    6 IF (N3.LE.3)
     *                  GO TO 10
    7 IF (EX1.GT.1.E-8)
     *                       GO TO 10
      K=3
      GO TO 27
   10 AL=EX1/P
      ALM=ABS(AL)
      AN=AL*SQRT(ALM*GM)
```

```
      GO TO (11,12,13,14),N1
   11 EL(1)=P
      GO TO 15
   12 EL(1)=1./AL
      GO TO 15
   13 EL(1)=-AL*GM
      GO TO 15
   14 EL(1)=PI2/AN
   15 W=0.
      IF (EX.GT.1.E-8)
     *                    W=ATAN3(F1,F2)
      GO TO (16,17,16,17),N2
   16 EL(2)=F1
      EL(3)=F2
      GO TO 19
   17 EL(2)=EX
      EL(3)=W
   19 EL(4)=CI
      IF (N2.LE.2)
     *              EL(4)=ATAN3(SI,CI)
      EL(5)=ATAN3(SOM,COM)
      IF (N3.NE.1)
     *              GO TO 20
      EL(6)=U
      RETURN
   20 IF (EX1.LE.1.E-8.AND.W.GT.PI2/2.)
     *                                      W=W-PI2
      GO TO (26,21,22,21,22),N3
   22 TEO=-W
      S=F2
      IF (1.+S.GT.1.E-10)
     *                    GO TO 23
      PRINT 100
  100 FORMAT('ABEL: ВОСХ. УЗЛА НЕТ! ВЫЧИСЛЯЕТСЯ ВРЕМЯ'
     *       ' ОТ ПЕРИЦЕНТРА')
   21 TEO=0.
      S=EX
   23 FI=U+W-TEO
      ARG=P/(1.+S)
      IF (EX1.GT.1.E-8)
     *                    GO TO 24
      IF (U-W.GT.PI2/2.)
     *                    FI=FI-PI2
      GO TO 25
   24 IF (FI.LT.0.)
     *              FI=FI+PI2
   25 DT=DTIME(GM,AL,ARG,AN,TEO,FI)
      IF (N3.GT.3)
     *              DT=DT+AN
      EL(6)=DT
   26 RETURN
   27 CALL PDIAGN('ABEL',K)
      END
```

ПОДПРОГРАММА ELAB(GM,NEL,EL,X)

Служит для вычисления вектора прямоугольных фазовых координат $\bar{x}$ по заданной совокупности элементов орбиты $\bar{y}$, определяемой числом $N_{эл}$.

Исходные данные: GM,NEL,EL(6)(μ,$N_{эл}$,$\bar{y}$).

Результат: X(6) ($\bar{x}$) .

Примечание. При неправильном задании элементов орбиты или числа $N_{эл}$ печатается соответствующий текст и счет прекращается.

Используемые подпрограммы и функции: NUMEL,ATAN3,KEPLER.

```
      SUBROUTINE ELAB(GM,NEL,EL,X)
      DIMENSION EL(6),X(6),R0(3),S0(3)
      DIMENSION N(3)
      EQUIVALENCE (N1,N(1)),(N2,N(2)),(N3,N(3))
      DATA PI2/6.28318530718/
      CALL NUMEL(NEL,N)
      GO TO (1,2,1,2),N2
    1 F1=EL(2)
      F2=EL(3)
      EX=SQRT(F1**2+F2**2)
      W=0.
      IF (EX.GT.1.E-8)
     -                         W=ATAN3(F1,F2)
      GO TO 3
    2 EX=EL(2)
      IF (EX.LT.0.)
     -                  GO TO 25
      W=EL(3)
      SW=SIN(W)
      CW=COS(W)
      F1=EX*SW
      F2=EX*CW
    3 CONTINUE
      EX1=(1.-EX)*(1.+EX)
   10 GO TO (11,12,13,14),N1
   11 P=EL(1)
      GO TO 16
   12 A=EL(1)
      GO TO 15
   13 IF (ABS(EL(1)).LE.1.E-12)
     -                                  GO TO 25
      A=-GM/EL(1)
      GO TO 15
   14 IF (EL(1).LE.0..OR.EX1.LE.1.E-8)
     -                                          GO TO 25
      A=((EL(1)/PI2)**2*GM)**(1./3.)
   15 P=A*EX1
```

```
16 IF (P.LE.0.)
   -                  GO TO 25
   IF (N2.GT.2)
   -                  GO TO 17
   IF (EL(4).LT.0.)
   -                         GO TO 25
   SI=SIN(EL(4))
   CI=COS(EL(4))
   GO TO 18
17 CI=EL(4)
   IF (ABS(CI).GT.1.)
   -                           GO TO 25
   SI=SQRT((1.-CI)*(1.+CI))
18 SOM=SIN(EL(5))
   COM=COS(EL(5))
   DT=EL(6)
   GO TO (19,20,21,20,21),N3
19 SU=SIN(EL(6))
   CU=COS(EL(6))
   GO TO 23
20 SU=SW
   CU=CW
   GO TO 22
21 SU=0.
   CU=1.
22 IF (N3.LE.3)
   -                  GO TO 23
   IF (EX1.LE.1.E-8)
   -                         GO TO 25
   IF (N1.EQ.1)
   -                  A=P/EX1
   DT=DT*A*SQRT(A/GM)
23 AR=P/(1.+F1*SU+F2*CU)
   AC=SQRT(GM*P)
   VR=AC/P*(F2*SU-F1*CU)
   VN=AC/AR
   RO(1)=COM*CU-SOM*SU*CI
   RO(2)=SOM*CU+COM*SU*CI
   RO(3)=SU*SI
   SO(1)=-COM*SU-SOM*CU*CI
   SO(2)=-SOM*SU+COM*CU*CI
   SO(3)=CU*SI
   DO 24 I=1,3
   X(I)=AR*RO(I)
24 X(I+3)=VR*RO(I)+VN*SO(I)
   IF (N3.EQ.1)
   -                  RETURN
   CALL KEPLER(GM,DT,X,X)
   RETURN
25 PRINT 100
100 FORMAT('ELAB: НЕВЕРНО ЗАДАНЫ ЭЛЕМЕНТЫ ОРБИТЫ '
   *          'ИЛИ ЧИСЛО NEL; СТОП')
   STOP
   END
```

ПОДПРОГРАММА ELEL(GM,NEL1,EL1,NEL2,EL2)

Служит для вычисления совокупности элементов орбиты $\bar{y}_2$, определяемой числом $N_{эл2}$, по заданной совокупности элементов $\bar{y}_1$, определяемой числом $N_{элI}$.

Исходные данные: GM,NEL1,EL1(6),NEL2(μ, $N_{элI}$, $\bar{y}_1$, $N_{эл2}$).

Результат: EL2(6)($\bar{y}_2$).

Используемые подпрограммы: ELAB,ABEL.

```
SUBROUTINE ELEL(GM,NEL1,EL1,NEL2,EL2)
DIMENSION EL1(6),EL2(6),X(6)
CALL ELAB(GM,NEL1,EL1,X)
CALL ABEL(GM,X,NEL2,EL2)
RETURN
END
```

ПОДПРОГРАММА PELOEL(GM,ELO,NEL,EL,PEL)

Служит для вычисления матрицы производных $P_{эл} = \frac{d\bar{y}_o}{d\bar{y}}$, где $\bar{y}_o$ – основная совокупность элементов орбиты, $\bar{y}$ – совокупность элементов, определяемая числом $N_{эл}$.

Исходные данные: GM,ELO(6),NEL,EL(6)($\mu,\bar{y}_o,N_{эл},\bar{y}$).

Результат: PEL(6,6)(PEL(I,J)=$P_{эл_{ij}}$).

Примечание. Если для заданной совокупности $\bar{y}$ матрица $P_{эл}$ не существует, печатается соответствующий текст и счет прекращается.

Используемые подпрограммы: NUMEL,PDIAGN.

```
      SUBROUTINE PELOEL(GM,ELO,NEL,EL,PEL)
      DIMENSION ELO(6),EL(6),PEL(6,6),Z(4)
      DIMENSION N(3)
      EQUIVALENCE (N1,N(1)),(N2,N(2)),(N3,N(3))
      DATA Z/1.,1.,-1.,1.5/
      DO 2 I=1,6
      DO 1 J=1,6
    1 PEL(I,J)=0.
    2 PEL(I,I)=1.
      IF (NEL.EQ.0)
     -                  RETURN
      CALL NUMEL(NEL,N)
      F1=ELO(2)
      F2=ELO(3)
      EX=SQRT(F1**2+F2**2)
      EX1=(1.-EX)*(1.+EX)
      K=2
      IF (N1*N3.NE.1.AND.ABS(EX1).LE.1.E-8)
     -                                              GO TO 31
   10 P=ELO(1)
      AI=ELO(4)
      U=ELO(6)
      IF (N1.EQ.1)
     -                 GO TO 12
      A=P/EX1
      PEL(1,1)=P/EL(1)/Z(N1)
      PEL(1,2)=-2.*A*EL(2)
      GO TO (11,12,11,12),N2
   11 PEL(1,3)=-2.*A*F2
   12 GO TO (14,13,14,13),N2
   13 SW=SIN(EL(3))
      CW=COS(EL(3))
```

```
      PEL(2,2)=SW
      PEL(3,2)=CW
      PEL(2,3)=F2
      PEL(3,3)=-F1
   14 IF (N2.LE.2)
     -                GO TO 16
      SI=SIN(AI)
      IF (ABS(SI).GT.1.E-10)
     -                             GO TO 15
      K=0
      GO TO 31
   15 PEL(4,4)=-1./SI
   16 IF (N3.EQ.1)
     -                RETURN
      SU=SIN(U)
      CU=COS(U)
      R=P/(1.+F1*SU+F2*CU)
      ROM=P/(1.+F2)
      C=SQRT(GM*P)
      GO TO (17,20,17,20),N2
   17 IF (EX.LE.1.E-8)
     -                   GO TO 18
      SW=F1/EX
      CW=F2/EX
      GO TO 20
   18 K=1
      GO TO (30,31,19,31,19),N3
   19 SW=0.
      CW=1.
   20 S1=(1.+P/R)/EX1*(SU*CW-CU*SW)
      IF (N3.GT.3)
     -                GO TO 21
      PEL(6,6)=C/R/R
      PEL(6,1)=-1.5*PEL(6,6)*EL(6)/EL(1)/Z(N1)
      IF (N1.EQ.1)
     -                S1=S1-3.*PEL(6,6)*EX*EL(6)/EX1
      GO TO 22
   21 PEL(6,6)=(P/R)**2/EX1/SQRT(EX1)
   22 GO TO (30,23,25,23,25),N3
   23 S3=1.
      GO TO (26,28,26,28),N2
   25 S1=S1+(ROM/R)**2*(2.+F2)/EX1*SW
      S2=ROM/P*(1.+ROM/R)*((1.-CU)*CW-SU*SW)
      S3=S2*EX
      GO TO (27,28,27,28),N2
   26 S2=1./EX
   27 PEL(6,2)=S1*SW+S2*CW
      PEL(6,3)=S1*CW-S2*SW
      RETURN
   28 PEL(6,2)=S1
      PEL(6,3)=S3
   30 RETURN
   31 CALL PDIAGN('PELOEL',K)
      END
```

ПОДПРОГРАММА PELEL(GM,NEL1,EL1,NEL2,EL2,PEL)

Служит для вычисления матрицы производных $P_{эл} = \frac{d\bar{y}_1}{d\bar{y}_2}$, где $\bar{y}_1, \bar{y}_2$ – совокупности элементов орбиты, определяемые числами $N_{элI}$, $N_{эл2}$.

Исходные данные: GM, NEL1, EL1(6), NEL2, EL2(6) (μ, $N_{элI}$, $\bar{y}_1$, $N_{эл2}$, $\bar{y}_2$).

Результат: PEL(6,6) (PEL(I,J)=$P_{элij}$).

Используемые подпрограммы: ELEL, PELOEL, MATIN1 (обращение матрицы), MUM.

```
      SUBROUTINE PELEL(GM,NEL1,EL1,NEL2,EL2,PEL)
      DIMENSION EL1(6),EL2(6),EL0(6),PEL(6,6),
     *          PEL1(6,6),PEL2(6,6),IND(6,2)
      CALL ELEL(GM,NEL1,EL1,0,EL0)
      CALL PELOEL(GM,EL0,NEL1,EL1,PEL1)
      CALL PELOEL(GM,EL0,NEL2,EL2,PEL2)
      CALL MATIN1(PEL1,6,6,6,0,IND,NER,DET)
      CALL MUM(PEL1,PEL2,PEL,6,6,6)
      RETURN
      END
```

ПОДПРОГРАММА PABEL(GM,X,NEL,EL,PM)

Служит для вычисления матрицы производных $P = \frac{d\bar{x}}{d\bar{y}}$, где совокупность элементов орбиты $\bar{y}$ определяется числом $N_{эл}$.

Исходные данные: GM,X(6),NEL,EL(6) $(\mu,\bar{x},N_{эл},\bar{y})$.

Результат: PM(6,6)(PM(I,J)=P_{ij}).

Используемые подпрограммы: ABEL,PELOEL,MUM.

```
      SUBROUTINE PABEL(GM,X,NEL,EL,PM)
      DIMENSION X(6),EL(6),EL0(6),PM(6,6),PEL(6,6),PM1(6,6)
      COMMON /BLRC8/AR,AC,C(3),C0(3)
      EQUIVALENCE (CI,C0(3))
      COMMON /BLSC4/SOM,COM,SU,CU
      EQUIVALENCE (P,EL0(1)),(F2,EL0(3)),(AI,EL0(4))
      CALL ABEL(GM,X,0,EL0)
      C1=-AR/P
      C2=AR*SU
      C3=AR**2/AC
      C4=-0.5/P
      C5=AC/P
      C6=(F2+CU)/P
      C7=-C5/AR
      DO 1 I=1,3
      PM(I,1)=X(I)/P
      PM(I,2)=C1*SU*X(I)
      PM(I,3)=C1*CU*X(I)
      PM(I,4)=C2*C0(I)
      PM(I,6)=C3*X(I+3)
      PM(I+3,1)=C4*X(I+3)
      PM(I+3,4)=C6*C(I)
    1 PM(I+3,6)=C7*X(I)
      PM(1,5)=-X(2)
      PM(2,5)=X(1)
      PM(3,5)=0.
      PM(4,5)=-X(5)
      PM(5,5)=X(4)
      PM(6,5)=0.
      PM(4,2)=-C5*COM
      PM(5,2)=-C5*SOM
      PM(6,2)=0.
      PM(4,3)=-C5*SOM*CI
      PM(5,3)= C5*COM*CI
      PM(6,3)=C5*SIN(AI)
      IF (NEL.EQ.0)
     *                    RETURN
      CALL PELOEL(GM,EL0,NEL,EL,PEL)
      CALL MUM(PM,PEL,PM1,6,6,6)
      DO 4 I=1,36
    4 PM(I)=PM1(I)
      RETURN
      END
```

ПОДПРОГРАММА PELAB(GM,NEL,EL,X,PM)

Служит для вычисления матрицы производных $P = \frac{d\bar{y}}{d\bar{x}}$, где совокупность элементов орбиты $\bar{y}$ определяется числом $N_{эл}$.

Исходные данные: GM,NEL,EL(6),X(6) (μ, $N_{эл}$, $\bar{y}$, $\bar{x}$).

Результат: PM(6,6)(PM(I,J)=P_{ij}).

Используемые подпрограммы: PABEL,MATIN1 (обращение матрицы).

```
SUBROUTINE PELAB(GM,NEL,EL,X,PM)
DIMENSION X(6),EL(6),PM(6,6),IND(6,2)
CALL PABEL(GM,X,NEL,EL,PM)
CALL MATIN1(PM,6,6,6,0,IND,NER,DET)
RETURN
END
```

ПОДПРОГРАММА DERIVE(GM,DT,X,XO,FI)

Служит для вычиоления матрицы производных $\Phi = \frac{d\bar{x}}{d\bar{x}_0}$ вторым способом.

Исходные данные: GM,DT,X(6),XO(6) (μ ,$t-t_0$,x,x_0).

Результат: FI(6,6)(FI(I,J)= Φ_{ij}).

Используемые подпрограммы: ABEL,PABEL,PELAB,MUM.

```
SUBROUTINE DERIVE(GM,DT,X,XO,FI)
DIMENSION XO(6),X(6),EL(6),FI(6,6),PM1(6,6),PM2(6,6)
CALL ABEL(GM,X,1,EL)
CALL PABEL(GM,X,1,EL,PM1)
EL(6)=EL(6)-DT
CALL PELAB(GM,1,EL,XO,PM2)
CALL MUM(PM1,PM2,FI,6,6,6)
RETURN
END
```

ВСПОМОГАТЕЛЬНЫЕ ПОДПРОГРАММЫ И ФУНКЦИИ

CS	- вычисление функций C(Y), S(Y) (см. п.2.I);
PCHOIC	- выбор векторов $\bar{p}_1, \bar{p}_2$ по заданному вектору $\bar{r}$ (см. п.2.2);
MATRA	- вычисление матрицы A (см. п.2.2);
MATRD	- вычисление матриц D^1 (см. п.2.3);
G	- вычисление компонент G_{ij} матрицы G (см. п.2.3);
NUMEL	- разбиение числа $N_{эл}$ на числа $\nu_1+1, \nu_2+1, \nu_3+1$ (см. п.3.I);
PDIAGN	- печать диагностики и остановка машины в случае, если при использовании заданной совокупности элементов орбиты $\bar{y}$ возникает особенность;
ATAN3	- вычисление arctg $\frac{x}{y}$ в диапазоне $[0,2\pi)$;
VMOD	- вычисление модуля N-мерного вектора A ;
NORM	- вычисление нормы AO и модуля U 3-мерного вектора A ;
SPR	- вычисление скалярного произведения N-мерных векторов A,B;
VPB	- вычисление векторного произведения AxB=C;
MUM	- вычисление произведения AB = C, где A,B,C - матрицы размеров LxM, MxN, LxN;
MUMV	- вычисление произведения AB = C, где A - матрица порядка N , B,C - N-мерные векторы;
TRM	- транспонирование матрицы A размеров MxN; результат - матрица B размеров NxM.

```
      SUBROUTINE CB(V,C,S)
      IF(ABS(V).GT.0.1)
     -                        GOTO 2
      AN=2.
      C=0.5
      S=1./6.
      CN=0.5
      SN=1./6.
    1 AN=AN+2.
      CN=-CN/(AN-1.)/AN*V
      SN=-SN/(AN+1.)/AN*V
      C=C+CN
      S=S+SN
      IF (ABS(CN)-1.E-13) 5,1,1
    2 X=SQRT(ABS(V))
      IF(V.LT.0.)
     -                 GOTO 3
      CX=COS(X)
      SX=SIN(X)
      GOTO 4
    3 EX=EXP(X)
      CX=0.5*(EX+EXP(-X))
      SX=EX-CX
    4 C=(1.-CX)/V
      S=(X-SX)/X/V
    5 RETURN
      END
```

```
      SUBROUTINE PCHOIC(R,V)
      DIMENSION R(3),V(3)
      COMMON /BLPCS/P1(3),P2(3),C(3),S1,S2,PARAB
      LOGICAL PARAB
      AR=VMOD(R,3)
      AV=VMOD(V,3)
      IF (ABS(SPR(R,V,3)).GE.0.9*AR*AV.OR.PARAB)
     -                                              GO TO 2
      DO 3 I=1,3
      P1(I)=R(I)/AR
    3 P2(I)=V(I)/AV
      RETURN
    2 K1=1
      K2=2
      IF (ABS(R(2)).GT.ABS(R(3)))
     -                                  K2=3
      IF (ABS(R(1)).GT.ABS(R(2)).AND.K2.EQ.3)
     -                                          K1=2
      IF (ABS(R(1)).GT.ABS(R(3)).AND.K2.EQ.2)
     -                                          K1=3
      DO 1 I=1,3
      P1(I)=0.
    1 P2(I)=0.
      P1(K1)=1.
      P2(K2)=1.
      RETURN
      END
```

```
      SUBROUTINE MATRA(GM,DT,R,V,A)
      DIMENSION R(3),V(3),A(6,6),U(3),W(3),PI(3),Z1(3),Z2(3),
     *          Z3(3),Z4(3)
      COMMON /BLPCS/P(3,2),C(3),S1,S2,PARAB
      LOGICAL PARAB
      RM=VMOD(R,3)
      GN=GM/RM/RM/RM
      DO 2 I=1,2
      DO 1 J=1,3
    1 PI(J)=P(J,I)
      CALL VPB(PI,R,U)
      CALL VPB(V,PI,W)
      CALL VPB(R,U,Z1)
      CALL VPB(V,W,Z2)
      CALL VPB(PI,C,Z3)
      CALL VPB(R,W,Z4)
      DO 2 J=1,3
      A(I,J)=W(J)
      A(I,J+3)=U(J)
      A(I+2,J)=GN*Z1(J)+Z2(J)
    2 A(I+2,J+3)=Z3(J)-Z4(J)
      IF (PARAB)
     -          GO TO 3
      C2=1.
      C3=2.
      C4=-3.*DT
      C1=C4*GN
      GO TO 5
    3 C2=SPR(C,C,3)-GM*RM
      C3=C2
      C4=-S1
      IF (DT.EQ.0.)
     -                C4=0.
      C1=GM*SPR(R,V,3)/RM*C4*GN
    5 DO 6 J=1,3
      A(5,J)=GN*R(J)
      A(5,J+3)=V(J)
      A(6,J)=C1*R(J)+C2*V(J)
    6 A(6,J+3)=C3*R(J)+C4*V(J)
      RETURN
      END
```

```
      SUBROUTINE MATRD(GM,DT,R,V,D,L)
      DIMENSION R(3),V(3),D(6,6),PS(3),VC(3),CR(3)
      COMMON /BLPCS/P(3,2),C(3),S1,S2,PARAB
      LOGICAL PARAB
      RM=VMOD(R,3)
      GN=GM/RM/RM/RM
      DO 1 I=1,36
    1 D(I)=0.
      GO TO (10,10,20,20,30,40),L
   10 D(1,5)=P(3,L)
      D(1,6)=-P(2,L)
      D(2,6)=P(1,L)
      DO 11 I=1,3
      DO 11 J=1,3
      IF (J.GT.I)
     -          D(J,I+3)=-D(I,J+3)
   11 D(I+3,J)=D(J,I+3)
      RETURN
```

```
20 L1=L-2
   RP=R(1)*P(1,L1)+R(2)*P(2,L1)+R(3)*P(3,L1)
   VP=V(1)*P(1,L1)+V(2)*P(2,L1)+V(3)*P(3,L1)
   DO 21 I=1,3
   DO 21 J=1,3
   W=0.
   IF (I.EQ.J)
  -                 W=1.
   U=R(I)*P(J,L1)+P(I,L1)*R(J)
   D(I,J)=-GN*U+RP*G(I,J,R,GN)
   D(I,J+3)=V(I)*P(J,L1)-2.*P(I,L1)*V(J)+VP*W
   D(J+3,I)=D(I,J+3)
21 D(I+3,J+3)=U-2.*RP*W
   RETURN
30 DO 32 I=1,3
   DO 31 J=1,3
31 D(I,J)=-G(I,J,R,GN)
32 D(I+3,I+3)=1.
   RETURN
40 U=3.*DT
   IF (PARAB)
  -                 GO TO 50
   DO 42 I=1,3
   DO 41 J=1,3
41 D(I,J)=U*G(I,J,R,GN)
   D(I,I+3)=1.
   D(I+3,I)=2.
42 D(I+3,I+3)=-U
   RETURN
50 CALL VPB(V,C,VC)
   CALL VPB(C,R,CR)
   R2=RM**2
   DO 51 I=1,3
51 PS(I)=-GN*R2*R(I)+VC(I)
   RP=SPR(R,PS,3)
   VP=SPR(V,PS,3)
   C1=S1
   C2=S2
   IF (DT.NE.0.)
                      GO TO 52
   C1=0.
   C2=0.
52 DO 53 J=1,3
   W1=U*VC(J)-C2*GN*R(J)+CR(J)
   W2=U*CR(J)-C2*V(J)+2.*R2*R(J)
   DO 53 I=1,3
   W=0.
   IF (I.EQ.J)
  -                 W=1.
   C3=R(I)*PS(J)
   C4=GN*R(I)
   D(I,J)=C1*G(I,J,R,GN)+V(I)*PS(J)-VP*W-C4*W1
   D(I,J+3)=RP*W-C3-C4*W2
   D(I+3,J)=RP*W+C3-V(I)*W1
53 D(I+3,J+3)=-C1*W-V(I)*W2
   RETURN
   END
```

```
      FUNCTION G(I,J,R,GN)
      DIMENSION R(3)
      G=3.*GN*R(I)*R(J)/SPR(R,R,3)
      IF (I.EQ.J)
     *               G=G-GN
      RETURN
      END


      SUBROUTINE NUMEL(NEL,N)
      DIMENSION N(3),NM(3)
      DATA NM/3,3,6/
      IF (NEL.LT.0)
     *                  NEL=-NEL
      N(1)=NEL/100
      N(2)=NEL/10-N(1)*10
      N(3)=NEL-N(1)*100-N(2)*10
      DO 1 I=1,3
      IF (N(I).LE.NM(I))
     *                           GO TO 1
      PRINT 100,I,N(I)
  100 FORMAT(' NUMEL : ПРИЗНАК N'I1'='I2' ЗАДАН НЕВЕРНО;'
     *          ' ПРИНИМАЕТСЯ = 0')
      N(I)=0
    1 N(I)=N(I)+1
      RETURN
      END


      SUBROUTINE PDIAGN(NSUBR,K)
      DIMENSION NAME(3,6)
      DATA NAME/'     КРУГОВАЯ    ',' ПАРАБОЛИЧЕСКАЯ',
     *         'НЕЭЛЛИПТИЧЕСКАЯ',' ПРЯМОЛИНЕЙНАЯ '/
      IF (K.NE.0)
     *               GO TO 1
      PRINT 100,NSUBR
  100 FORMAT(A7': НУЛЕВОЕ НАКЛОНЕНИЕ! СТОП')
      STOP
    1 PRINT 101,NSUBR,(NAME(I,K),I=1,3)
  101 FORMAT(A7': '2A6,A3' ОРБИТА! СТОП')
      STOP
      END


      FUNCTION ATAN3(X,Y)
      U=ATAN2(X,Y)
      IF(U.LT.0.)
     *               U=U+6.28318530718
      ATAN3=U
      RETURN
      END
```

```
      FUNCTION VMOD(A,N)
      DIMENSION A(N)
      U=0
      DO 1 I=1,N
    1 U=U+A(I)**2
      VMOD=SQRT(U)
      RETURN
      END
```

```
      SUBROUTINE NORM(A,A0,U)
      DIMENSION A(3),A0(3)
      U=VMOD(A,3)
      DO 1 I=1,3
    1 A0(I)=A(I)/U
      RETURN
      END
```

```
      FUNCTION SPR(A,B,N)
      DIMENSION A(N),B(N)
      C=0.
      DO 1 I=1,N
    1 C=C+A(I)*B(I)
      SPR=C
      RETURN
      END
```

```
      SUBROUTINE VPB(A,B,C)
      DIMENSION A(3),B(3),C(3)
      C(1)=A(2)*B(3)-A(3)*B(2)
      C(2)=A(3)*B(1)-A(1)*B(3)
      C(3)=A(1)*B(2)-A(2)*B(1)
      RETURN
      END
```

```
      SUBROUTINE MUM(A,B,C,L,M,N)
      DIMENSION A(L,M),B(M,N),C(L,N)
      LOGICAL NOTSUM
      NOTSUM=.TRUE.
      GO TO 1
      ENTRY MUMS
      NOTSUM=.FALSE.
    1 DO 2 I=1,L
      DO 2 J=1,N
      IF (NOTSUM)
     -              C(I,J)=0.
      DO 2 K=1,M
    2 C(I,J)=C(I,J)+A(I,K)*B(K,J)
      RETURN
      END
```

```
      SUBROUTINE MUMV(A,B,C,N)
      DIMENSION A(N,N),B(N),C(N
      DO 1 I=1,N
      C(I)=0
      DO 1 J=1,N
    1 C(I)=C(I)+A(I,J)*B(J)
      RETURN
      END
```

```
      SUBROUTINE TRM(A,B,M,N)
      DIMENSION A(M,N),B(N,M)
      DO 1 I=1,M
      DO 1 J=1,N
    1 B(J,I)=A(I,J)
      RETURN
      END
```

ЛИТЕРАТУРА

I. Бэттин Р. Наведение в космосе. М., "Машиностроение", 1966.

2. Бахшиян Б.Ц., Суханов А.А. Об изохронных производных первого и второго порядка в задаче двух тел. Космические исследования, 1978, 16, вып. 4.

©

055(02)2 Отпечатано в ИКИ АН СССР

Т-12736 Подписано к печати 13.11.78

Заказ 1691 Тираж 120 Объем 2,3 уч.-изд.л.