ИНСТИТУТ АСТРОНОМИИ
РОССИЙСКОЙ АКАДЕМИИ НАУК

ВЫЧИСЛЕНИЕ ЦЕЛЕУКАЗАНИЙ ДЛЯ НАБЛЮДЕНИЯ МАЛЫХ ТЕЛ, ОПАСНО СБЛИЖАЮЩИХСЯ С ЗЕМЛЁЙ.

Модели, алгоритмы, примеры вычислений.

Научно-технический отчёт

на 49 страницах

Краткое содержание.
Представлено описание алгоритмов прогноза положений малых тел Солнечной системы и алгоритмов вычисления целеуказаний для наблюдений астероидов, сближающихся с Землёй. Описание снабжено комментариями и вычислительными примерами.

Москва – 2007

Содержание

1 Постановка задачи **4**

2 Алгоритмы операций с матрицами и векторами **7**

2.1 Вычисление матриц поворота . . . 7

2.2 Умножение матрицы на вектор . . . 8

2.3 Умножение матрицы на матрицу . . . 9

2.4 Вычисление транспонированной матрицы . . . 10

2.5 Файл unforfun.h – прототипы функций . . . 11

3 Алгоритмы преобразования даты **12**

3.1 Календарная дата и модифицированный юлианский день . . . 12

3.2 Файл unfordat.h – прототипы функций . . . 12

3.3 Контрольный пример . . . 13

4 Две шкалы равномерного времени **14**

4.1 Вычисление земного времени . . . 14

4.2 Вычисление барицентрического динамического времени . . . 15

5 Прецессия и нутация **16**

5.1 Вычисление параметров прецессии . . . 16

5.2 Вычисление матрицы прецессии . . . 17

5.3 Вычисление угла наклона эклиптики . . . 18

5.4 Вычисление фундаментальных аргументов . . . 18

5.5 Вычисление параметров нутации . . . 19

5.6 Вычисление матрицы нутации . . . 21

5.7 Прототипы функций . . . 22

5.8 Контрольный пример . . . 23

6 Звёздное время **24**

6.1 Всемирное координированное время . . . 24

6.2 Всемирное время . . . 24

6.3 Гринвичское среднее звёздное время 25
6.4 Гринвичское истинное звёздное время 26
6.5 Матрица вращения Земли 27
6.6 Файл unforsit.h – прототипы функций 27
6.7 Контрольный пример 27

7 Алгоритмы вычисления матриц преобразований 28

7.1 От средней экваториальной системы координат к небесной . . . 28
7.2 От небесной к истинной экваториальной системе координат . . 28
7.3 От истинной экваториальной системы координат к небесной . . 29
7.4 Преобразование из небесной системы координат в земную . . . 29
7.5 Преобразование из земной системы координат в небесную . . . 30
7.6 Файл unforrom.h – прототипы функций 31
7.7 Контрольный пример 31

8 Алгоритмы преобразования вектора состояния 32

8.1 Вектор состояния 32
8.2 Файл unforkep.h – прототипы функций 34
8.3 Контрольный пример 35

9 Положения планет 36

9.1 Файл unephjpl.h – прототипы функций 36
9.2 Контрольный пример 37

10 Интегрирование уравнений движения 38

10.1 Правые части 38
10.2 Алгоритм интегрирования 38
10.3 Шаг численного интегрирования 39

11 Алгоритм вычисления целеуказаний 40

11.1 Геодезические координаты на поверхности Земли 40
11.2 Топоцентрическая экваториальная система координат 41
11.3 Топоцентрическая горизонтальная система координат 42

11.4 Условия видимости 43
11.5 Время суток 45
11.6 Видимая звёздная величина 45
11.7 Азимут, высота, часовой угол и склонение 46
11.8 Контрольный пример 47

12 Результаты расчётов 48

Литература 49

Список таблиц

1 Апофис 13 апреля 2029 года 6
2 Поправка к шкале всемирного времени 14
3 Выбор шага интегрирования 39
4 Условия видимости 48

1 Постановка задачи

В предлагаемом отчёте представлен алгоритм вычисления целеуказаний для наблюдения малых тел Солнечной системы. Алгоритм включает в себя формулы преобразований между различными системами отсчёта и различными шкалами времени.

Алгоритм использован для детального изучения обстоятельств будущих сближений малого тела 99942 Апофис с Землёй.

Земная система отсчёта жёстко связана с телом Земли и вращается вместе с Землёй. В земной системе координат заданы положения пунктов наблюдений.

Небесная система отсчёта задана положением экватора, фиксированного на стандартную эпоху J2000.0 (2000.0, январь, 1.5). Ось абсцисс направлена в среднюю точку весеннего равноденствия, соответствующую стандартной эпохе.

Преобразование из небесной системы отсчёта (C) в земную (T) выполняется по формуле:

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix}_{\mathrm{T}} = R_2(-x_p) \cdot R_1(-y_p) \cdot \mathbf{R} \cdot \mathbf{N} \cdot \mathbf{P} \cdot \begin{pmatrix} x \\ y \\ z \end{pmatrix}_{\mathrm{C}},$$

где

$\mathbf{P}$ – матрица прецессии,

$\mathbf{N}$ – матрица нутации,

$\mathbf{R} = R_3(S_\oplus)$ – матрица вращения Земли,

$R_1(-y_p)$ – матрица поворота вокруг оси абсцисс по часовой стрелки на эмпирическое значение координаты полюса Земли y_p ,

$R_2(-x_p)$ – матрица поворота вокруг оси ординат по часовой стрелки на эмпирическое значение координаты полюса Земли x_p .

Аргументом матриц прецессии и нутации является время в шкале барицентрического динамического времени TDB. Матрица вращения Земли образована поворотом вокруг оси аппликат на угол, равный значению истинного звёздного времени $S_\oplus$. Истинное звёздное время является функцией всемирного времени UT1. Всемирное время, в свою очередь, образуется добавлением поправки ΔUT к значению всемирного координированного времени UTC.

Величины ΔUT, x_p, y_p называются параметрами вращения Земли и могут быть определены только в результате наблюдений. В настоящее время численные значения координат полюса не превосходят по модулю 0.5 секунды дуги, а абсолютная величина поправки всемирного времени не может стать более 0.9 секунды времени. Во многих случаях столь малыми значениями параметров можно пренебречь.

Центром O топоцентрической горизонтальной системы координат является пункт наблюдений на поверхности Земли. Плоскость XOY пересекает линию горизонта. Ось OZ перпендикулярна горизонтальной плоскости. Ось OX направлена к северу.

Геодезические координаты пункта наблюдений на поверхности Земли, прямоугольные $\bar{X}_p$, $\bar{Y}_p$, $\bar{Z}_p$ и сферические φ_p, λ_p, необходимы для преобразований между земной и топоцентрической системами координат.

Начальные параметры движения астероида Апофис известны в виде оскулирующих кеплеровских элементов орбиты в гелиоцентрической эклиптической системе отсчёта.

10.04.2007 – эпоха оскулирующих элементов орбиты,

$T_0 = 54200.0$ – эпоха в модифицированных юлианских днях,

$a = 0.92226144$ – большая полуось в астрономических единицах,

$e = 0.19105939$ – эксцентриситет орбиты,

$i = 3.331313$ – угол наклонения орбиты в градусах,

$\Omega = 204.45925$ – долгота восходящего узла в градусах (J2000.0),

$\omega = 126.385488$ – аргумент перигелия в градусах (J2000.0),

$M(T_0) = 307.363034$ – средняя аномалия в градусах (J2000.0),

$H = 19.20$ – абсолютная звёздная величина,

$G = 0.15$ – параметр наклона.

Числовые значения угловых величин даны в системе эклиптики, фиксированной на стандартную эпоху J2000.0 (2000.0, январь, 1.5).

Точность оскулирующих элементов орбиты очень высока. Средняя квадратическая погрешность положения малого тела по направлению движения составляет 0.15 секунды дуги. Оценка скорости увеличения этой величины равна 0.00054 секунды дуги в день.

Погрешность по направлению можно представить в виде ошибки начального численного значения средней аномалии $\sigma_{M(T_0)} = 0\overset{\prime\prime}{.}15$. Скорость увеличения погрешности по направлению обусловлена ошибкой в начальном значении среднего движения $\sigma_n = 3 \cdot 10^{-9}$ радиан в день. Такой величине ошибки соответствует оценка средней квадратической погрешности начального значения большой полуоси $\sigma_a = 20$ километров.

Для прогнозирования положений малого тела применяется метод численного интегрирования. Уравнения движения записаны в гелиоцентрической экваториальной системе отсчёта. Центр системы находится в центре Солнца. Ось абсцисс и ось ординат расположены в плоскости небесного экватора, фиксированного на эпоху J2000.0 (2000.0, январь, 1.5). Ось абсцисс направлена в точку весеннего равноденствия, соответствующую стандартной эпохе.

В апреле 2029 года малое тело Апофис пройдёт на небольшом расстоянии от поверхности Земли. В табл. 1 представлено изменение параметров сближения во времени. Минимальное расстояние обозначено δr, AU – астрономическая единица. В колонке m записана видимая звёздная величина. Заметно резкое изменение числовых значений элементов орбиты астероида.

Таблица 1: Апофис 13 апреля 2029 года

UTC	a (AU)	e	i (°)	δr (AU)	δr (km)	m
19^h00^m	0.9315	0.2132	3.76	0.00050	74719.8	3.7
20^h00^m	0.9431	0.2176	3.83	0.00037	55486.5	3.4
21^h00^m	0.9739	0.2219	3.83	0.00027	40585.5	3.4
21^h30^m	1.0018	0.2231	3.72	0.00025	36795.4	3.7
21^h40^m	1.0127	0.2233	3.65	0.00024	36339.7	3.9
21^h50^m	1.0238	0.2235	3.58	0.00024	36327.6	4.2
22^h00^m	1.0348	0.2235	3.51	0.00025	36759.4	4.5
23^h00^m	1.0838	0.2211	3.05	0.00031	47031.4	7.0

2 Алгоритмы операций с матрицами и векторами

2.1 Вычисление матриц поворота

Для преобразований между различными системами отсчёта используют операцию поворота системы координат и операцию переноса начальной точки.

Поворот системы координат выполняется с помощью матрицы поворота. Положительным направлением называется поворот против часовой стрелки.

Дан

угол α, действительное число, единица измерений – радианы.

Вычислить

матрицы поворота на угол α вокруг осей OX, OY, OZ.

Алгоритм вычислений.

Матрицу поворота на угол α вокруг оси OX обозначим $R_x(\alpha)$:

$$R_x(\alpha) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & \sin\alpha \\ 0 & -\sin\alpha & \cos\alpha \end{pmatrix}.$$

Матрицу поворота на угол α вокруг оси OY обозначим $R_y(\alpha)$:

$$R_y(\alpha) = \begin{pmatrix} \cos\alpha & 0 & -\sin\alpha \\ 0 & 1 & 0 \\ \sin\alpha & 0 & \cos\alpha \end{pmatrix}.$$

Матрицу поворота на угол α вокруг оси OZ обозначим $R_z(\alpha)$:

$$R_z(\alpha) = \begin{pmatrix} \cos\alpha & \sin\alpha & 0 \\ -\sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Программная реализация алгоритма.

Модуль **unforfun.c**

void forrotmatr

2.2 Умножение матрицы на вектор

Результатом умножения матрицы на вектор является вектор.

Даны

матрица

$$R = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix},$$

вектор

$$\vec{r} = \begin{pmatrix} r_1 \\ r_2 \\ r_3 \end{pmatrix}.$$

Вычислить

Произведение

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} \cdot \begin{pmatrix} r_1 \\ r_2 \\ r_3 \end{pmatrix}.$$

Алгоритм вычислений.

$$R \cdot \vec{r} = \begin{pmatrix} a_{11}r_1 + a_{12}r_2 + a_{13}r_3 \\ a_{21}r_1 + a_{22}r_2 + a_{23}r_3 \\ a_{31}r_1 + a_{32}r_2 + a_{33}r_3 \end{pmatrix}.$$

Программная реализация алгоритма.

Модуль **unforfun.c**

void multmatrvec

2.3 Умножение матрицы на матрицу

Результатом произведения двух матриц является новая матрица.

Даны

матрица

$$R_a = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix},$$

матрица

$$R_b = \begin{pmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{pmatrix}.$$

Вычислить

произведение

$$\begin{pmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} \cdot \begin{pmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{pmatrix}.$$

Алгоритм вычислений.

$$c_{11} = a_{11}b_{11} + a_{12}b_{21} + a_{13}b_{31},$$

$$c_{12} = a_{11}b_{12} + a_{12}b_{22} + a_{13}b_{32},$$

$$c_{13} = a_{11}b_{13} + a_{12}b_{23} + a_{13}b_{33},$$

$$c_{21} = a_{21}b_{11} + a_{22}b_{21} + a_{23}b_{31},$$

$$c_{22} = a_{21}b_{12} + a_{22}b_{22} + a_{23}b_{32},$$

$$c_{23} = a_{21}b_{13} + a_{22}b_{23} + a_{23}b_{33},$$

$$c_{31} = a_{31}b_{11} + a_{32}b_{21} + a_{33}b_{31},$$

$$c_{32} = a_{31}b_{12} + a_{32}b_{22} + a_{33}b_{32},$$

$$c_{33} = a_{31}b_{13} + a_{32}b_{23} + a_{33}b_{33}.$$

2.4 Вычисление транспонированной матрицы

Результатом транспонирования матрицы является новая матрица. Произведение матрицы поворота на транспонированную матрицу равно единичной матрице.

Дана

матрица

$$R_a = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix}.$$

Вычислить

матрицу

$$R_b = \begin{pmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{pmatrix},$$

транспонированную по отношению к матрице R_a.

Алгоритм вычислений.

$$\begin{aligned} b_{11} &= a_{11}, \quad b_{12} = a_{21}, \quad b_{13} = a_{31}, \\ b_{21} &= a_{12}, \quad b_{22} = a_{22}, \quad b_{23} = a_{32}, \\ b_{31} &= a_{13}, \quad b_{32} = a_{23}, \quad b_{33} = a_{33}. \end{aligned}$$

Программная реализация алгоритма.

Модуль **unforfun.c**

void transpmatr

2.5 Файл unforfun.h – прототипы функций

```
double  sqr ( double a ) ;

double  frac ( double a ) ;

double  datan2 ( double sa , double ca ) ;

double  atandegree ( double s , double c  ) ;

double  scalarmult ( double a[4] , double b[4] ) ;

void forrotmatr ( int iaxis , double angle , double matr[4][4] ) ;

void tomultmatr ( double ma[4][4], double mb[4][4], double mc[4][4] ) ;

void multmatrvec ( double p[4][4], double v[4], double vp[4] ) ;

void transpmatr ( double ma[4][4], double mb[4][4] ) ;
```

3 Алгоритмы преобразования даты

3.1 Календарная дата и модифицированный юлианский день

Одним из важных параметров алгоритмов является время. Время может быть выражено в самых разнообразных формах. В вычислениях наиболее удобная форма — сквозная нумерация суток. Счёт юлианских дней идёт от 1 января 4713 года до нашей эры. Юлианские сутки начинаются в полдень.

Модифицированные юлианские дни образованы простым вычитанием из значения юлианских дней числа 2400000.5. Модифицированные юлианские сутки начинаются в гринвичскую полночь.

Сквозная нумерация суток, необходимая для вычислений внутри всех алгоритмов, совершенно неприемлема для задания исходных данных и выдачи результатов расчётов. Для этих целей используют понятную и доступную всем форму календарных дат.

Задача *первого алгоритма* состоит в преобразовании от календарной даты к модифицированному юлианскому дню.

Задача *второго алгоритма* состоит в преобразовании модифицированного юлианского дня к календарной дате.

Программная реализация алгоритмов.

Модуль **unfordat.c**

double fromdatetomjd

void transmjdtodate

Алгоритмы дают правильный результат на интервале календарных дат со 2 марта 1900 года по 27 февраля 2100 года.

3.2 Файл unfordat.h – прототипы функций

```
double fromdatetomjd ( int nday , int nmonth , int nyear ,
                       int nhour , int nminute , double second ) ;
void fromsecparttohourminsec ( double secpart ,
                               int *inthour , int *intmin ,
                               double *second ) ;
```

```
void transmjdtodate ( double tmjd ,
                      int *nday , int *nmonth , int *nyear ,
                      int *nhour , int *nminute ,
                      double *second ) ;
```

3.3 Контрольный пример

```
 input
 moscow decret winter time
         29      day
          2      month
       2036      year
          5      hour
         45      minute
          0.000   second

 output
 moment in modified julian day
      64752.11458333  day with part in utc scale

 input
 moment in modified julian day
      64752.11458333  day with part in utc scale

 output
 moscow decret winter time
         29      day
          2      month
       2036      year
          5      hour
         45      minute
          0.000  second
```

4 Две шкалы равномерного времени

4.1 Вычисление земного времени

На поверхности Земли и в околоземном пространстве пользуются равномерной шкалой земного времени TT.

Дано:

текущая дата и время в шкале всемирного координированного времени (UTC) в форме

год	—	целое число,
месяц	—	целое число,
день	—	целое число,
час	—	целое число,
минута	—	целое число,
секунда	—	действительное число.

Вычислить

соответствующий момент земного времени TT в модифицированных юлианских днях.

Алгоритм вычислений.

С помощью алгоритма разд. 3.1 на с. 12 надо вычислить модифицированную юлианскую дату mjd, соответствующую заданной текущей дате и времени.

Таблица 2: Поправка к шкале всемирного времени

mjd	ΔT	дата
49534.0	61.184	1994 07 01
50083.0	62.184	1996 01 01
50630.0	63.184	1997 07 01
51178.0	64.184	1999 01 01
53736.0	65.184	2006 01 01

Далее на основе табл. 2 надо узнать поправку ΔT в секундах времени.

Момент в шкале земного времени равен

$$TT = \mathrm{mjd} + \frac{\Delta T}{86400}\,.$$

Программная реализация алгоритма.

Модуль **unfortim.c**

void transutctotdt

4.2 Вычисление барицентрического динамического времени

В пределах Солнечной системы пользуются равномерной шкалой барицентрического динамического времени TDB.

В этой шкале вычисляются положения Луны, Солнца и параметры прецессии и нутации.

Дан

Момент в шкале земного времени TT, выраженный в модифицированных юлианских днях.

Вычислить

Соответствующее значение барицентрического динамического времени TDB, выраженное в модифицированных юлианских днях.

Алгоритм вычислений.

Вспомогательные переменные:

d — интервал времени в юлианских столетиях, от стандартной эпохи J2000.0, начало которой соответствует 1.5 января 2000 года, до текущего момента,

g — приближённое значение аргумента перигелия Земли в радианах.

$$d = \frac{TT - 51544.5}{36525.0}\,,$$

$$g = 0.017453 \cdot (357.258 + 35999.050 \cdot d)\,,$$

$$TDB = TT + \frac{0.001658 \cdot \sin(g + 0.0167 \cdot \sin g)}{86400}\,.$$

Программная реализация алгоритма.

Модуль **unfortim.c**

void transtdttotdb

5 Прецессия и нутация

5.1 Вычисление параметров прецессии

Параметрами прецессии называются три угловые переменные: $\zeta_a(t)$, $\theta_a(t)$, $z_a(t)$. Аргументом t для вычисления параметров прецессии является барицентрическое динамическое время TDB, выраженное в модифицированных юлианских днях.

С помощью параметров прецессии вычисляют матрицу прецессии для преобразования от системы координат, соответствующей стандартной эпохе J2000.0, в систему подвижного экватора и мгновенной эклиптики, соответствующую заданной эпохе.

Дан

момент времени t в шкале барицентрического динамического времени TDB, выраженный в модифицированных юлианских днях.

Вычислить

параметры прецессии, то есть числовые значения трёх угловых переменных $\zeta_a(t)$, $\theta_a(t)$, $z_a(t)$.

Алгоритм вычислений.

В постоянной $r_s = 4.848136811095 \cdot 10^{-6}$ хранится число для перевода дуговых секунд в радианную меру.

Вспомогательная переменная t_c содержит время в юлианских столетиях, прошедшее от стандартной эпохи J2000.0, начало которой соответствует 1.5 января 2000 года.

$$t_c = \frac{t - 51544.5}{36525.0}$$

Числовые значения равны

$$\begin{aligned}
\zeta_a(t) &= r_s \cdot (2306.2181 + (0.30188 + 0.017998 \cdot t_c) \cdot t_c) \cdot t_c, \\
\theta_a(t) &= r_s \cdot (2004.3109 - (0.42665 + 0.041833 \cdot t_c) \cdot t_c) \cdot t_c, \\
z_a(t) &= r_s \cdot (2306.2181 + (1.09468 + 0.018203 \cdot t_c) \cdot t_c) \cdot t_c.
\end{aligned}$$

Единицы измерений.

Параметры прецессии выражены в радианах.

Программная реализация алгоритма.

Модуль **unforpnm.c**

void clcprecangles

5.2 Вычисление матрицы прецессии

Матрица прецессии необходима для преобразования от системы координат, соответствующей стандартной эпохе J2000.0, в систему среднего подвижного экватора и мгновенной эклиптики, соответствующую заданной эпохе.

Дан

момент времени t в шкале барицентрического динамического времени TDB.

Вычислить

матрицу прецессии $\mathbf{P}(t)$.

Алгоритм вычислений.

Матрица прецессии суть произведение трёх матриц поворота:

$$\mathbf{P}(t) = R_z(-z_a(t)) \cdot R_y(\theta_a(t)) \cdot R_z(-\zeta_a(t)).$$

На момент времени t вычисляются параметры прецессии $\zeta_a(t)$, $\theta_a(t)$, $z_a(t)$, выраженные в радианах (алгоритм разд. 5.1 на с. 16).

С помощью алгоритма (разд. 2.1 на с. 7) последовательно вычисляются три матрицы поворота

$$R_z(-\zeta_a(t)), \quad R_y(\theta_a(t)), \quad R_z(-z_a(t)).$$

Далее необходимо использовать алгоритм умножения матриц:

$$\mathbf{P}(t) = R_z(-z_a(t)) \cdot R, \qquad R = R_y(\theta_a(t)) \cdot R_z(-\zeta_a(t)),$$

где R — вспомогательная матрица.

Программная реализация алгоритма.

Модуль **unforpnm.c**

void clcprecangles

void clcprecmatr

5.3 Вычисление угла наклона эклиптики

Числовое значение угла наклона мгновенной эклиптики к среднему подвижному экватору $\varepsilon(t)$, заданное на момент барицентрического динамического времени t, необходимо для вычисления матрицы нутации.

Дан

момент времени t в шкале барицентрического динамического времени.

Вычислить

числовое значение $\varepsilon(t)$ угла наклона эклиптики к среднему экватору.

Алгоритм вычислений.

$$\varepsilon(t) = r_s \cdot (84381.448 - (46.815 + (0.0059 - 0.001813 \cdot t_c) \cdot t_c) \cdot t_c)\,.$$

Программая реализация алгоритма.

Модуль **unfornut.c**

double togetepsmean

5.4 Вычисление фундаментальных аргументов

Фундаментальными аргументами называются следующие угловые переменные:

λ — средняя долгота Луны,

l — средняя аномалия Луны,

l' — средняя аномалия Солнца,

F — средний аргумент широты Луны,

D — разность средних долгот Луны и Солнца,

Фундаментальные аргументы являются функциями барицентрического динамического времени t и используются для вычисления параметров нутации и в теориях движения Луны и Солнца.

Дан

момент времени t в шкале барицентрического динамического времени.

Вычислить

числовые значения фундаментальных аргументов.

Алгоритм вычислений.

$$
\begin{aligned}
r_g &= 0.017453292519943296\,,\\
t_c &= \frac{t-51544.5}{36525}\,,\\
\lambda(t) &= r_g\cdot(218.31643250\\
&\quad + (481267.8812772222-(0.00161167-0.00000528\cdot t_c)\cdot t_c)\cdot t_c)\,,\\
l(t) &= r_g\cdot(134.96298139\\
&\quad + (477198.8673980556+(0.00869722+0.00001778\cdot t_c)\cdot t_c)\cdot t_c)\,,\\
l'(t) &= r_g\cdot(357.52772333\\
&\quad + (35999.05034-(0.00016028+0.00000333\cdot t_c)\cdot t_c)\cdot t_c)\,,\\
F(t) &= r_g\cdot(93.27191028\\
&\quad + (483202.0175380555-(0.00368250-0.00000306\cdot t_c)\cdot t_c)\cdot t_c)\,,\\
D(t) &= r_g\cdot(297.85036306\\
&\quad + (445267.11148-(0.00191417-0.00000528\cdot t_c)\cdot t_c)\cdot t_c)\,.
\end{aligned}
$$

Единицы измерений.

Фундаментальные аргументы выражены в радианах.

Программная реализация алгоритма.

Модуль **unfornut.c**

void clcfundarg

5.5 Вычисление параметров нутации

Параметрами нутации называются следующие угловые переменные:

$\Delta\psi$ — нутация в долготе,

$\Delta\varepsilon$ — нутация в наклоне.

Параметры нутации являются функциями барицентрического динамического времени и необходимы для вычисления матрицы нутации. Числовое значение нутации в долготе используется при вычислении истинного звёздного времени.

Дан момент времени t в шкале TDB.

Вычислить

параметры нутации $\Delta\psi(t)$, $\Delta\varepsilon(t)$.

Алгоритм вычислений.

На момент времени t с помощью алгоритма разд. 5.4 со с. 18 необходимо вычислить фундаментальные аргументы

$$\lambda(t),\ \ l(t),\ \ l'(t),\ \ F(t),\ \ D(t).$$

Далее надо использовать формулы:

$$\begin{aligned}
\Delta\psi(t) = r_s \cdot \big[&(-17.1996 - 0.01742 \cdot t_c) \cdot \sin(\lambda - F) \\
&+ (0.2062 + 0.00002 \cdot t_c) \cdot \sin(2\lambda - 2F) \\
&+ 0.0046 \cdot \sin(\lambda - 2l + F) \\
&+ 0.0011 \cdot \sin(2l - 2F) \\
&- (1.3187 + 0.00016 \cdot t_c) \cdot \sin(2\lambda - 2D) \\
&+ (0.1426 - 0.00034 \cdot t_c) \cdot \sin l' \\
&- (0.0517 - 0.00012 \cdot t_c) \cdot \sin(2\lambda + l' - 2D) \\
&+ \ \ldots\big]\,,
\end{aligned}$$

$$\begin{aligned}
\Delta\varepsilon(t) = r_s \cdot \big[&(9.2025 + 0.00089 \cdot t_c) \cdot \cos(\lambda - F) \\
&- (0.0895 - 0.00005 \cdot t_c) \cdot \cos(2\lambda - 2F) \\
&- 0.0024 \cdot \cos(\lambda - 2l + F) \\
&+ (0.5736 - 0.00031 \cdot t_c) \cdot \cos(2\lambda - 2D) \\
&+ (0.0054 - 0.00001 \cdot t_c) \cdot \cos l' \\
&+ (0.0224 - 0.00006 \cdot t_c) \cdot \cos(2\lambda + l' - 2D) \\
&+ \ldots\big]\,,
\end{aligned}$$

где

$$r_s = 4.848136811095 \cdot 10^{-6}\,,$$
$$t_c = \frac{t - 51544.5}{36525}\,.$$

Единицы измерений.

Параметры нутации выражены в радианах.

Программная реализация алгоритма.

Модуль **unfornut.c**

void clcnut

5.6 Вычисление матрицы нутации

Матрица нутации необходима для преобразования от системы координат, соответствующей среднему подвижному экватору, в систему координат, соответствующей истинному экватору.

Дан

момент времени t в шкале барицентрического динамического времени.

Вычислить

матрицу нутации $\mathbf{N}(t)$.

Алгоритм вычислений.

Матрица нутации суть произведение трёх матриц поворота:

$$\mathbf{N}(t) = R_x(-\varepsilon - \Delta\varepsilon) \cdot R_z(-\Delta\psi) \cdot R_x(\varepsilon).$$

Вычислить угол наклона мгновенной эклиптики к среднему подвижному экватору $\varepsilon(t)$ (алгоритм разд. 5.3 на с. 18).

На момент времени t в шкале барицентрического динамического времени вычислить параметры нутации $\Delta\psi(t)$, $\Delta\varepsilon(t)$ (алгоритм разд. 5.5 на с. 19).

С помощью алгоритма (разд. 2.1 на с. 7) последовательно вычислить три матрицы поворота:

$$R_x(-\varepsilon - \Delta\varepsilon), \;\; R_z(-\Delta\psi), \;\; R_x(\varepsilon).$$

Далее необходимо использовать алгоритм умножения матриц

$$R = R_z(-\Delta\psi(t)) \cdot R_x(\varepsilon(t)),$$
$$\mathbf{N}(t) = R_x(-\varepsilon(t) - \Delta\varepsilon(t)) \cdot R,$$

где R — вспомогательная матрица.

Программная реализация алгоритма.

Модуль **unforpnm.c**

void clcnutmatr

5.7 Прототипы функций

Файл **unforpnm.h** :

```
void clcprecangles ( double epoch , double tmjd ,
                     double *dzita0 , double *teta , double *zet ) ;

void clcprecmatr ( double epoch , double tmjd , double precmatr[4][4] ) ;

void clcnutmatr ( double deltapsi , double deltaeps , double epsmean ,
                  double matrnut[4][4] ) ;
```

Файл **unfornut.h** :

```
double togetepsmean ( double tmjd ) ;

void clcfundarg ( double tmjd , double fundarg[6] ) ;

void clcnut ( double tmjd , double  *deltapsi , double *deltaeps ) ;
```

5.8 Контрольный пример

```
input

59152.0  moment in modified julian day

output

precession matrix  P

 0.999987104372264  -0.004657817791984  -0.002023813872801
 0.004657817791642   0.999989152296765  -0.000004713477259
 0.002023813873587  -0.000004713139788   0.999997952075499

nutation   matrix  N

 0.999999995979347   0.000082275361072   0.000035666114213
-0.000082275089649   0.999999996586437  -0.000007611504096
-0.000035666740330   0.000007608569633   0.999999999334997

output

result of multiplication P*N

 0.999987555756883  -0.004575558874333  -0.001988112764180
 0.004575543743875   0.999989532071017  -0.000012158772523
 0.001988147585937   0.000003061924296   0.999998023627948
```

6 Звёздное время

6.1 Всемирное координированное время

Момент всемирного координированного времени UTC соответствует моменту московского декретного времени, уменьшенному на три часа.

Моменту всемирного координированного времени UTC соответствует модифицированная юлианская дата UTC_{mjd}.

Дано:

текущая дата и время в шкале всемирного координированного времени (UTC) в форме

год	—	целое число,
месяц	—	целое число,
день	—	целое число,
час	—	целое число,
минута	—	целое число,
секунда	—	действительное число.

Вычислить

соответствующий текущему моменту времени момент всемирного координированного времени в форме модифицированной юлианской даты UTC_{mjd}.

Алгоритм вычислений.

Для вычислений надо использовать алгоритм разд. 3.1 на с. 12.

Единицы измерений.

Величина UTC_{mjd} измеряется в юлианских днях.

6.2 Всемирное время

Всемирное время UT1 связано со всемирным координированным временем UTC формулой

$$\mathrm{UT1} = \mathrm{UTC} + \Delta\mathrm{UT}\,.$$

Единицы измерений.

Поправка всемирного времени $\Delta\mathrm{UT}$ измеряется в секундах.

6.3 Гринвичское среднее звёздное время

Гринвичское среднее звёздное время $S^m_\oplus$ является функцией всемирного времени UT1.

Разность $\Delta\mathrm{UT} = \mathrm{UT1} - \mathrm{UTC}$ всемирного времени и всемирного координированного времени может быть определена только на основе наблюдений.

Даны

момент всемирного координированного времени UTC_{mjd} и числовое значение поправки $\Delta\mathrm{UT}$, выраженное в секундах времени.

Вычислить

гринвичское среднее звёздное время.

Алгоритм вычислений.

Переменная t_c содержит текущее значение всемирного времени в модифицированных юлианских днях:

$$t_c = \mathrm{UTC}_{mjd} + \frac{\Delta\mathrm{UT}}{86400}\,.$$

Функция $\mathrm{INT}(x)$ вычисляет целую часть действительного числа x:

$$T_U = \frac{\mathrm{INT}(\,t_c) - 51544.5}{36525}\,,$$

где T_U – время, отсчитываемое в юлианских столетиях по 36525 суток в системе всемирного времени UT1 от эпохи 2000, январь 1, 12^h UT1 (MJD51544.5).

$$\begin{aligned} S^m_\oplus \text{ в } 0^h \text{ UT1 } &= 1.753368559233266 + (628.3319706888409 \\ &+ (6.770714 \cdot 10^{-6} - 4.51 \cdot 10^{-10} \cdot T_U) \cdot T_U) \cdot T_U\,, \end{aligned}$$

Промежуток звёздного времени от 0^h UT1 до момента, соответствующего моменту всемирного времени UT1, равен

$$S^m_\oplus = S^m_\oplus \text{ в } 0^h \text{ UT1 } + r \cdot \Big[\,t_c - \mathrm{INT}(\,t_c)\Big],$$

где

$$r = 6.300388098984891 + (3.707456 \cdot 10^{-10} - 3.707 \cdot 10^{-14} \cdot T_U) \cdot T_U.$$

Гринвичское среднее звёздное время $S^m_\oplus$ измеряется в радианах.

6.4 Гринвичское истинное звёздное время

Гринвичское среднее звёздное время $S_{\oplus}^{m}$ является функцией всемирного времени UT1.

Разность ΔUT = UT1 - UTC всемирного времени и всемирного координированного времени может быть определена только на основе наблюдений.

Гринвичское истинное звёздное время $S_{\oplus}$ есть угол от истинной точки весеннего равноденствия до гринвичского меридиана, отсчитываемый вдоль истинного экватора.

Даны

момент времени UTC и числовое значение поправки ΔUT.

Вычислить

гринвичское истинное звёздное время $S_{\oplus}$.

Алгоритм вычислений.

На основе алгоритма разд. 6.3 на с. 25 с помощью исходных данных UTC, ΔUT определить числовое значение гринвичского среднего звёздного времени $S_{\oplus}^{m}$.

Далее на момент времени $t = \mathrm{UTC}_{mjd}$ надо вычислить угол наклона мгновенной эклиптики к среднему подвижному экватору $\varepsilon(t)$ (алгоритм разд. 5.3 на с. 18) и параметр нутации $\Delta\psi(t)$ (алгоритм разд. 5.5 на с. 19).

Формула для вычисления гринвичского истинного звёздного времени гласит

$$S_{\oplus} \;=\; S_{\oplus}^{m} + \Delta\psi\,\cos\varepsilon.$$

Единицы измерений.

Гринвичское истинное звёздное время $S_{\oplus}$ измеряется в радианах.

Программная реализация алгоритма.

Модуль **unforsit.c**

double togetgmstime

double togetgastime

6.5 Матрица вращения Земли

Матрица вращения Земли $\mathbf{R} = R_z(S_\oplus)$ является матрицей поворота вокруг оси OZ на угол, равный значению истинного звёздного времени $S_\oplus$.

6.6 Файл unforsit.h – прототипы функций

```
double  togetgmstime ( double tinutc , double deltaut ) ;
double  togetgastime ( double tinutc , double deltaut ) ;
void earthrotmatr ( double utc , double dut ,  double rz[4][4] ) ;
```

6.7 Контрольный пример

```
input
moscow decret winter time
         7      day
        12      month
      1999      year
         5      hour
        45      minute
         0.000  second
universal time correction ut1-utc
         0.384  second of time
output
moment in modified julian day
     51519.11458333  day with part in utc scale
moment in terrestrial time scale
     51519.11532620  day with part in tt scale
          84381.481  obliquity in second of arc
            -14.939  d(psi)    in second of arc
greenwich mean sidereal time
     2.036644850536  in radian
greenwich true sidereal time
     2.036578399019  in radian
earth rotation matrix
    -0.449121697258     0.893470593278     0.000000000000
    -0.893470593278    -0.449121697258     0.000000000000
     0.000000000000     0.000000000000     1.000000000000
```

7 Алгоритмы вычисления матриц преобразований

7.1 От средней экваториальной системы координат к небесной

Дана

матрица прецессии $\mathbf{P}(t)$.

Вычислить

матрицу преобразования от средней подвижной системы координат к небесной системе координат $\mathbf{M}_p$.

Алгоритм вычислений.

Матрица $\mathbf{M}_p$ является транспонированной по отношению к матрице прецессии $\mathbf{P}(t)$.

Для вычисления надо воспользоваться алгоритмом транспонирования матрицы (разд. 2.4 на с. 10).

Программная реализация алгоритма.

Модуль **unforrom.c**

void frommeantofixequm

7.2 От небесной к истинной экваториальной системе координат

Матрица перехода между небесной системой координат и истинной экваториальной системой является произведением матрицы нутации $\mathbf{N}(t)$ на матрицу прецессии $\mathbf{P}(t)$.

Даны

матрицы прецессии $\mathbf{P}(t)$ и нутации $\mathbf{N}(t)$.

Вычислить

матрицу преобразования от небесной системы координат к истинной экваториальной системе координат $\mathbf{M}_{np}$.

Алгоритм вычислений.

Для вычисления надо воспользоваться алгоритмом разд. 2.3 на с. 9.

$$\mathbf{M}_{np} = \mathbf{N}(t) \cdot \mathbf{P}(t).$$

Программная реализация алгоритмов.

Модуль **unforrom.c**

void frommeantofixequm

void fromfixtotrueequm

7.3 От истинной экваториальной системы координат к небесной

Матрица перехода от истинной экваториальной системы координат к небесной системе координат является транспонированной по отношению к матрице перехода от небесной системы координат к истинной экваториальной.

Дана

матрица $\mathbf{M}_{np}$ (разд. 7.2 на с. 28).

Вычислить

матрицу преобразования от истинной экваториальной системы координат к небесной системе координат $\mathbf{M}_{pn}$.

Алгоритм вычислений.

Матрица $\mathbf{M}_{pn}$ является транспонированной по отношению к матрице $\mathbf{M}_{np}$.

Для вычисления надо воспользоваться алгоритмом транспонирования матрицы (разд. 2.4 на с. 10).

Программная реализация алгоритма.

Модуль **unforrom.c**

void fromtruetofixequm

7.4 Преобразование из небесной системы координат в земную

Матрицы перехода от небесной к земной системе координат вычисляется как произведение матрицы вращения Земли, матрицы нутации, матрицы прецессии.

Даны

матрица прецессии $\mathbf{P}(t)$ (разд. 5.2 на с. 17), матрица нутации $\mathbf{N}(t)$ (разд. 5.6 на с. 21)

матрица вращения Земли $R_z(S_\oplus)$ (разд. 6.5 на с. 27)

Вычислить

матрицу перехода от небесной к земной системе координат $\mathbf{M}_{ct}$.

Алгоритм вычислений.

Необходимо воспользоваться алгоритмом умножения матриц (разд. 2.3 на с. 9)

$$R = \mathbf{N}(t) \cdot \mathbf{P}(t),$$
$$\mathbf{M}_{ct} = R_z(S_\oplus) \cdot R,$$

где R – вспомогательная матрица.

Программная реализация алгоритма.

Модуль **unforrom.c**

void fromtruetofixequm

void fromfixtoterram

7.5 Преобразование из земной системы координат в небесную

Дана

Матрица преобразования из небесной системы координат в земную $\mathbf{M}_{ct}$.

Вычислить

матрицу преобразования из земной системы координат в небесную $\mathbf{M}_{tc}$.

Алгоритм вычислений.

Матрица $\mathbf{M}_{tc}$ равна матрице, транспонированной по отношению к матрице $\mathbf{M}_{ct}$.

Для вычисления надо воспользоваться алгоритмом транспонирования матрицы (разд. 2.4 на с. 10).

Программная реализация алгоритмов.

Модуль **unforrom.c**

void frommeantofixequm

void fromfixtotrueequm

void fromtruetofixequm

void fromfixtoterram

void fromterratofixm

7.6 Файл unforrom.h – прототипы функций

```
void frommeantofixequm ( double tc , double pf[4][4] ) ;
void fromfixtotrueequm ( double tc , double pn[4][4] ) ;
void fromtruetofixequm ( double tc , double pn[4][4] ) ;
void fromfixtoterram ( double utc , double dut ,
                       double *tdb ,double ptm[4][4] ) ;
void fromterratofixm ( double ptm[4][4] , double pzm[4][4] ) ;
```

7.7 Контрольный пример

```
from terrestrial to celestial system
input
moment in modified julian day
     51519.11458333  day with part in utc scale
universal time correction ut1-utc
         0.384  second of time
moment in terrestrial time scale
     51519.11532620  day with part in tt scale
moment in ephemeris time scale
     51519.11532619  day with part in tdb scale
greenwich mean sidereal time
     2.036644850536  in radian
greenwich true sidereal time
     2.036578399019  in radian
from celestial to terrestrial system matrix
    -0.449194954530     0.893433765159     0.000009932203
    -0.893433764506    -0.449194953704    -0.000044796575
    -0.000035561277    -0.000028996161     0.999999998947
output
from terrestrial to celestial system matrix
    -0.449194954530    -0.893433764506    -0.000035561277
     0.893433765159    -0.449194953704    -0.000028996161
     0.000009932203    -0.000044796575     0.999999998947
```

8 Алгоритмы преобразования вектора состояния

8.1 Вектор состояния

Вектор состояния малой планеты включает в себя следующие компоненты: начальную дату:

день — целое число,

месяц — целое число,

год — целое число;

набор оскулирующих кеплеровских элементов:

a — величину большой полуоси орбиты в астрономических единицах,

e — величину эксцентриситета орбиты,

i — величину угла наклонения орбиты в градусах,

Ω — величину долготы восходящего узла орбиты в градусах,

ω — величину аргумента перигея орбиты в градусах,

M_0 — величину средней аномалии орбиты в градусах.

Параметры i, Ω, ω даны в гелиоцентрической эклиптической системе координат. Числовое значение элемента M_0 относится к начальной дате.

При переходе от кеплеровских элементов к декартовым координатам и скоростям используют три угловые переменные: эксцентрическую аномалию E, истинную аномалию v и аргумент широты $u = v + \omega$. Средняя и эксцентрическая аномалии связаны между собой трансцедентным уравнением Кеплера

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

В формулах преобразования используется гелиоцентрическая гравитационная постоянная fM_S, измеряемая в $\mathrm{AU}^3/\mathrm{day}^2$, размерность координат – астрономическая единица, скоростей – астрономическая единица в сутки. Масса измеряется в единицах массы Солнца. Числовое значение величины fM_S равно квадрату гравитационной постоянной Гаусса

$$k = 0.01720209895.$$

Вычисления по формулам эллиптической орбиты дадут вектор положения $\vec{r}_{ecl}$ и вектор скорости $\dot{\vec{r}}_{ecl}$ малой планеты относительно Солнца в эклиптической системе.

Вектор подожения $\vec{r}_{ecl}$ имеет координаты

$$x_{ecl},\ \ y_{ecl},\ \ z_{ecl}.$$

Вектор скорости $\dot{\vec{r}}_{ecl}$ имеет координаты

$$\dot{x}_{ecl},\ \ \dot{y}_{ecl},\ \ \dot{z}_{ecl}.$$

Вектор положения астероида $\vec{r}$ в гелиоцентрической экваториальной системе имеет координаты x, y, z, а вектор скорости определён составляющими $\dot{x}$, $\dot{y}$, $\dot{z}$.

Переход от эклиптической к экваториальной системе выполняется по формулам

$$\begin{aligned}
x &= +x_{ecl}, \\
y &= +y_{ecl}\cdot\cos\varepsilon_A - z_{ecl}\cdot\sin\varepsilon_A, \\
z &= +y_{ecl}\cdot\sin\varepsilon_A + z_{ecl}\cdot\cos\varepsilon_A, \\
\dot{x} &= +\dot{x}_{ecl}, \\
\dot{y} &= +\dot{y}_{ecl}\cdot\cos\varepsilon_A - \dot{z}_{ecl}\cdot\sin\varepsilon_A, \\
\dot{z} &= +\dot{y}_{ecl}\cdot\sin\varepsilon_A + \dot{z}_{ecl}\cdot\cos\varepsilon_A.
\end{aligned}$$

Числовое значение угла наклона эклиптики к экватору ε_A является одной из астрономических постоянных и вычисляется по формуле

$$\varepsilon_A = 23^\circ 26' 21''\!.448.$$

Программная реализация алгоритмов.

Модуль **unforkep.c**,

double keplerequation — решение уравнения Кеплера с помощью итерационного метода Ньютона,

void fromkeplertocart — переход от кеплеровских элементов орбиты к декартовым векторам положения и скорости,

void fromcarttokepler — переход от декартовых векторов положения и скорости к значениям кеплеровских элементов орбиты.

8.2 Файл unforkep.h – прототипы функций

```
/* for minor planet
   MeanMotion   : mean motion  n  in radian per day
   semi-major axis in AU
   Sqr(GaussConst)  in AU**3/(day**2*SolarMass)
   from the formula
   n^2*a^3=k^2
   or  n=k/sqrt(a^3) where  k = 0.01720209895 - Gauss const */

double fromaxistomot ( double a ) ;    /* semimajor axis in AU */

/* kepler equation   M=E-e*sin(E)
   to solve with the use Newton iteration method
   if  fi(x)=0  then iteration may be possible
   x_(n+1)=f(x_n)  where  f(x)=x-fi(x)/(d/dx(fi(x))) */

double keplerequation ( double eleme , double meananom ) ;

/* output   ecliptical system
   posvar 1,2,3 : a in AU, e,  inclination in degree
   angvar 1,2,3 : the ascending node in degree
                : the argument of perigei in degree
                : the mean anomaly in degree */

void fromcarttokepler ( double start ,                          /*  input */
                        double pos[4] , double vel[4] ,         /*  input */
                        double finish ,                         /*  input */
                        double posvar[4] , double angvar[4] ) ; /* output */

/* output   equatorial system
   position pos 1,2,3  : x,y,z in AU
   velosity vel 1,2,3  : vx,vy,vz in AU/day */
```

```
void fromkeplertocart ( double start ,                /*  input */
                        double posvar[4] ,            /*  input */
                        double angvar[4] ,            /*  input */
                        double finish ,               /*  input */
                        double pos[4] , double vel[4] ) ; /* output */
```

8.3 Контрольный пример

```
initial elements
ecliptic heliocentric
     53000.00000000  epoch in mjd
         1.91997795  a semimajor axis   AU
         0.43460482  e eccentricity
          11.878789  i inclination degree
         171.418697  O ascending node deg.
          26.433709  o arg.perihelium deg.
         335.308176  M mean anomaly degree

coordinate and velosity
equator heliocentric
     54000.00000000  moment in mjd
       -1.088981018  x  coordinate   in AU
        0.386338321  y  coordinate   in AU
        0.114106614  z  coordinate   in AU
    -0.002909948721  vx velosity in AU/day
    -0.018262604782  vy velosity in AU/day
    -0.003687289170  vz velosity in AU/day

the same elements
ecliptic heliocentric
     53000.00000000  epoch in mjd
         1.91997795  a semimajor axis   AU
         0.43460482  e eccentricity
          11.878789  i inclination degree
         171.418697  O ascending node deg.
          26.433709  o arg.perihelium deg.
         335.308176  M mean anomaly degree
```

9 Положения планет

9.1 Файл unephjpl.h – прототипы функций

```
double  factormus[12] ;

double  factormuz[12] ;

double tiniteph,tlasteph ;

long  nofbyte,ibytemax ;

int  coefile, coebyte;

double  coefeph[745] ;

void toobtainfmus ( void ) ;

void clcposvel ( int  nplanet ,   /* input */
                 double  tintdb , /* input */
                 double  pos[4] ,
                 double  vel[4] ) ; /* output */

int toopencoefile ( void ) ; /* all variables are external */

void toreadephemeris ( double  tintdb ) ;

void clcephearthmoon ( double  tintdb , /* input */
                       double  posearth[4] ,
                       double  velearth[4] , /* output */
                       double  posmoon[4] ,
                       double  velmoon[4] ) ; /* output */

void earths ( double  tb ,            /*  input */
              double  x[4] ,
              double  v[4] ) ; /* output */
```

9.2 Контрольный пример

```
a try to open ephemeris file   0
the first record ephemeris file
       16.00       4208557.870944580994000            -0.000000049758453
the first moment and the last moment in ephemeris file
           16.00      104496.00

the moment to read ephemeris file     51550.00

 2       -0.7124583436473       -0.1497006396575       -0.0224414714450 AU
 2        0.0039069435935       -0.0180772977203       -0.0083796368453 AU/day

 3       -0.2779139100353        0.8646414950798        0.3750900188822 AU
 3       -0.0168203846191       -0.0044149810328       -0.0019135407775 AU/day

 4        1.3849915395478        0.0746548334154       -0.0031070951592 AU
 4       -0.0001616090105        0.0137836542532        0.0063267005983 AU/day

 5        3.9687941271309        2.7661484980882        1.0890153139375 AU
 5       -0.0046159401899        0.0058381325777        0.0026148831116 AU/day

 6        6.3756485427397        6.1913509059278        2.2828517150820 AU
 6       -0.0043004612336        0.0035085223832        0.0016340850254 AU/day

 7       14.4394987113804      -12.4954001100514       -5.6768979306772 AU
 7        0.0026805086789        0.0024578177110        0.0010385445137 AU/day

 8       16.8191802798003      -22.9735014098319       -9.8219137207224 AU
 8        0.0025836465122        0.0016630478881        0.0006163741824 AU/day

10       -0.2770524516587        0.8622660573984        0.3741307547728 AU
10       -0.0162906217746       -0.0042223682427       -0.0018847973637 AU/day

11       -0.0071095289382       -0.0026793654791       -0.0009376115142 AU
11        0.0000054239976       -0.0000067173997       -0.0000030164210 AU/day
```

10 Интегрирование уравнений движения

10.1 Правые части

Алгоритм вычисления правых частей дифференциальных уравнений движения **реализован** в модуле

unequras.c

void planetsforces

void clcrighthandcoor

Прототипы функций:

```
void planetsforces ( double teph ,        /* input moment */
                     double curp[4] ,     /* input position */
                     double forcec[4] ) ; /* output forces */

void clcrighthandcoor ( double ut ,         /* input */
                        double rc[4] ,      /* input */
                        double force[4] ) ; /* output */
```

10.2 Алгоритм интегрирования

Для численного интегрирования дифференциальных уравнений движения астероидов выбран метод Эверхарта 15-го порядка.

Алгоритм интегрирования дифференциальных уравнений движения **реализован** в модуле **uneverim.c**.

Прототипы функций:

```
void everinit (void) ;
void evercoefnullo (void) ;
void clcvectfrompolinom ( double dstep ,
                          double zeror[4] , double zerov[4] ,
                          double vectr[4] , double vectv[4] ) ;
void posvelevergrator ( double stepfull ,
                        double *tbeg , double zeror[4] , double zerov[4] ,
                        double *tend , double vectr[4] , double vectv[4] ) ;
```

10.3 Шаг численного интегрирования

В алгоритмах численного интегрирования дифференциальных уравнений движения величина шага по времени определяется экспериментальным путём. Существует одно соображение общего характера: шаг должен быть таким, чтобы на выбранном интервале вариации правых частей уравнений могли быть аппроксимированы полиномом по времени.

В задаче о движении малых тел Солнечной системы учитывается притяжения Солнца, больших планет и Луны. Период обращения Луны меньше одного месяца, шаг численного интегрирования не может превышать этой величины.

Для оценки величины шага численного интегрирования были выполнены пять вариантов расчёта. Начальный вектор состояния был один и тот же во всех вариантах. Числовые значения шага составляли 0.5 суток, 1 сутки, 2 суток, 5 и 10 суток. На конце интервала интегрирования, через 100 лет, выполнялось сравнение кеплеровских элементов орбиты. Результаты эксперимента представлены в табл. 3. Начальный вектор состояния соответствует параметрам движения астероида Икар.

Таблица 3: Выбор шага интегрирования

шаг	a (AU)	e	i (°)	Ω (°)	ω (°)
0.5	1.0778311	0.827372	22.62660	87.43274	31.97429
1.0	1.0778311	0.827372	22.62660	87.43274	31.97429
2.0	1.0778311	0.827372	22.62660	87.43274	31.97429
5.0	1.0778312	0.827372	22.62661	87.43274	31.97429
10.0	1.0784705	0.827371	22.62512	87.43337	31.99429

Для численного интегрирования уравнений движения был выбран постоянный шаг, равный 1 суткам. При сближениях малого тела с большими планетами до расстояний, меньших 1.5 миллионов километров величина шага интегрирования принимается равной 0.1 суток. Такая величина шага обеспечивает сохранение точности вычислений при сближениях с планетами до расстояний в 20 раз меньших расстояния от Земли до Луны.

11 Алгоритм вычисления целеуказаний

11.1 Геодезические координаты на поверхности Земли

Обозначим координаты пункта наблюдений в земной опорной системе, заданные на какую-либо дату, через $\bar{X}$, $\bar{Y}$, $\bar{Z}$.

Периодические изменения координат, обусловленные влиянием приливов упругой Земли и океанических приливов, всегда менее одного метра. Максимальные вековые смещения обсерваторий за десять лет также не превосходят одного метра. Отличия координат в системе истинного экватора от земной опорной системы вследствие движения полюсов не достигают трёх десятков метров. Во многих случаях такими отклонениями можно пренебречь.

Геодезические широта φ, долгота λ и высота пункта H с помощью экваториального радиуса Земли

$$a_e = 6378140.0 \text{ метров}$$

и сжатия Земли

$$f = 0.00335281 = \frac{1}{298.257}$$

связаны с координатами $\bar{X}, \bar{Y}, \bar{Z}$ формулами

$$\begin{aligned}
\bar{X} &= (G + H) \cdot \cos\varphi \cdot \cos\lambda, \\
\bar{Y} &= (G + H) \cdot \cos\varphi \cdot \sin\lambda, \\
\bar{Z} &= (G \cdot (1 - e^2) + H) \cdot \sin\varphi, \\
G &= \frac{a_e}{\sqrt{1 - e^2 \cdot \sin^2\varphi}}, \\
e^2 &= 2 \cdot f - f^2.
\end{aligned}$$

Для преобразования от прямоугольных координат к геодезическим вычислим долготу

$$P = \sqrt{\bar{X}^2 + \bar{Y}^2}, \quad \sin\lambda = \frac{\bar{Y}}{P}, \quad \cos\lambda = \frac{\bar{X}}{P},$$

и воспользуемся методом последовательных приближений:

начальное значение

$$\frac{H}{G} = 0,$$

далее достаточно трёх итераций

$$
\begin{aligned}
&Q = \sqrt{\left[\bar{Z} \cdot \left(1 + \frac{H}{G}\right)\right]^2 + \left[P \cdot \left(1 - e^2 + \frac{H}{G}\right)\right]^2},\\
&\sin\varphi = \bar{Z} \cdot \frac{1 + \dfrac{H}{G}}{Q},\\
&\cos\varphi = P \cdot \frac{1 - e^2 + \dfrac{H}{G}}{Q},\\
&G = \frac{a_e}{\sqrt{1 - e^2 \cdot \sin^2\varphi}},\\
&\frac{H}{G} = \frac{P}{G \cdot \cos\varphi} - 1.
\end{aligned}
$$

11.2 Топоцентрическая экваториальная система координат

Центром топоцентрической экваториальной системы координат является пункт наблюдений на поверхности Земли.

Кругом склонений называют дугу большого круга, соединяющего северный и южный полюсы мира и точку на небесной сфере. Круг склонений, пересекающий истинный экватор в точке юга, называют первым меридианом места наблюдения.

Для первой экваториальной системы используют сферические координаты произвольной точки на небесной сфере: δ – склонение, t_h – часовой угол. Для второй экваториальной системы используют сферические координаты произвольной точки на небесной сфере: δ – склонение, α – прямое восхождение.

Склонение δ измеряется в угловой мере вдоль дуги, отсчитываемой от экватора по кругу склонений. Северный полюс имеет склонение $\delta = +90^\circ$. Южный полюс имеет склонение $\delta = -90^\circ$. Часовой угол t_h отсчитывается по часовой стрелке вдоль экватора от первого меридиана до точки пересечения экватора с кругом склонений. Прямое восхождение α отсчитывается против часовой стрелки вдоль экватора от истинной точки весеннего равноденствия до пересечения с кругом склонений.

Часовой угол t_h, прямое восхождение α, истинное звёздное время $S_\oplus$ и долгота места наблюдения λ связаны формулой:

$$\begin{aligned} t_h &= s_l - \alpha, \\ s_l &= S_\oplus + \lambda, \end{aligned}$$

где s_l – местное звёздное время.

11.3 Топоцентрическая горизонтальная система координат

Центром O топоцентрической горизонтальной системы координат является пункт наблюдений на поверхности Земли. Плоскость XOY пересекает линию горизонта. Ось OZ перпендикулярна горизонтальной плоскости. Ось OX направлена к северу.

Геодезические координаты пункта наблюдений на поверхности Земли, прямоугольные $\bar{X}_b$, $\bar{Y}_b$, $\bar{Z}_b$ и сферические φ_b, λ_b, необходимы для преобразований между земной и топоцентрической системами координат.

Пусть в земной системе координат (разд. 1 на с. 4) задан вектор положения объекта

$$x_c, \;\; y_c, \;\; z_c,$$

привязанный к центру Земли.

Этот же вектор в земной системе отсчёта, но вычисляемый относительно пункта наблюдений, имеет координаты:

$$x_b = x_c - \bar{X}_b, \;\; y_b = y_c - \bar{Y}_b, \;\; z_b = z_c - \bar{Z}_b.$$

Преобразование из земной системы к топоцентрической горизонтальной системе координат может быть выполнено с помощью трёх поворотов против часовой стрелки. С помощью первого поворота вокруг оси OZ на угол λ_b ось OX будет направлена в точку, пересекающую меридиан пункта наблюдений. Второй поворот вокруг нового направления оси OY на угол $90^\circ - \varphi_b$, где φ_b – геодезическая широта, переведёт ось OZ в положение, перпендикулярное горизонтальной плоскости. Третьим поворотом вокруг нового направления оси OZ на угол 180° ось OX, лежащая в плоскости горизонта, окажется направленной на север.

Обозначим через $\mathbf{M}_{tq}$ матрицу преобразования от земной системы координат к топоцентрической. Формула для вычисления имеет вид

$$\mathbf{M}_{tq} = R_z(180^\circ) \cdot R_y(90^\circ - \varphi_b) \cdot R_z(\lambda_b).$$

В топоцентрической системе координат геодезический азимут A объекта наблюдений отсчитывается от направления на север по часовой стрелке, $0^\circ \leqslant A < 360^\circ$. Угол высоты h отсчитывается от линии горизонта против часовой стрелки, $0^\circ < h \leqslant 90^\circ$. Для объектов, оказавшихся ниже линии горизонта, угол высоты h принимает отрицательные значения: $-90^\circ \leqslant h \leqslant 0^\circ$.

11.4 Условия видимости

При определении условий видимости необходимо учитывать время распространения света. Изображение, зафиксированное на приёмнике излучения в момент времени t, содержит информацию о положении объекта в более ранний момент времени t', момент отражения солнечного света объектом.

Дан

момент времени t.

Известен алгоритм

расчёта положения малого тела на произвольный момент времени t'. Координаты объекта $x(t')$, $y(t')$, $z(t')$ вычислены в гелиоцентрической экваториальной системе отсчёта.

Найти

положение объекта относительно пункта наблюдений в топоцентрической горизонтальной системе координат на заданный момент времени t.

Алгоритм расчёта целеуказаний состоит из последовательных преобразований вычисленного положения объекта.

С помощью алгоритма разд. 9 (с.36) на момент времени t вычислим прямоугольные гелиоцентрические экваториальные координаты Земли X_E, Y_E, Z_E. В нулевом приближении на момент времени $t' = t$ найдём координаты объекта $x(t')$, $y(t')$, $z(t')$ в гелиоцентрической экваториальной системе отсчёта.

Далее по формуле

$$\rho = \sqrt{(x(t') - X_E(t))^2 + (y(t') - Y_E(t))^2 + (z(t') - Z_E(t))^2}$$

можно получить оценку расстояния малого тела от центра Земли и вычислить в первом приближении момент времени t':

$$t' = t - \frac{\rho}{c}, \quad c \text{ – скорость света.}$$

Вычисление координат объекта $x(t')$, $y(t')$, $z(t')$ в гелиоцентрической экваториальной системе отсчёта, расстояние малого тела от Земли ρ и оценку момента времени t' следует повторить один или два раза.

Видимые геоцентрические координаты объекта в небесной системе отсчёта определены формулами

$$x_f = x(t') - X_E(t), \;\; y_f = y(t') - Y_E(t) \;\; z_f = z(t') - Z_E(t).$$

Величины x_f, y_f, z_f с помощью матрицы M_{ct} (разд. 7.4 на с. 29) следует преобразовать к положениям $x_c(t)$, $y_c(t)$, $z_c(t)$ в земной системе отсчёта:

$$\begin{pmatrix} x_c \\ y_c \\ z_c \end{pmatrix} = M_{ct} \cdot \begin{pmatrix} x_f \\ y_f \\ z_f \end{pmatrix}.$$

Далее выполним операцию переноса центра системы координат в **пункт наблюдений**:

$$x_b = x_c - \bar{X}_b, \;\; y_b = y_c - \bar{Y}_b, \;\; z_b = z_c - \bar{Z}_b,$$

а с помощью матрицы $\mathbf{M}_{tq}$ и алгоритма умножения матрицы на вектор (разд. 2.2 на с. 8) преобразуем положение объекта в топоцентрические прямоугольные координаты $x_q(t)$, $y_q(t)$, $z_q(t)$:

$$\begin{pmatrix} x_q \\ y_q \\ z_q \end{pmatrix} = M_{tq} \cdot \begin{pmatrix} x_b \\ y_b \\ z_b \end{pmatrix}.$$

Если выполняется условие $z_q(t) < 0$, то объект находится вне зоны видимости, и целеуказания не вычисляются.

Если $z_q(t) > 0$, то объект находится в поле зрения данного пункта наблюдений.

11.5 Время суток

Условия освещённости на пункте наблюдений можно определить следующим образом.

С помощью алгоритма разд. 7.5 (с.30) вычислить матрицу $\mathbf{M}_{tc}$ для преобразования из земной системы координат в небесную.

С помощью матрицы $\mathbf{M}_{tc}$ и алгоритма умножения матрицы на вектор (с.8) преобразовать вектор положения пункта наблюдений к координатам $\bar{X}_q(t)$, $\bar{Y}_q(t)$, $\bar{Z}_q(t)$ в небесной системе отсчёта:

$$\begin{pmatrix} \bar{X}_q \\ \bar{Y}_q \\ \bar{Z}_q \end{pmatrix} = M_{tc} \cdot \begin{pmatrix} \bar{X}_b \\ \bar{Y}_b \\ \bar{Z}_b \end{pmatrix} .$$

С помощью алгоритма разд. 9 (с.36) на момент времени t вычислить прямоугольные координаты Солнца

$$X_S = -X_E, \;\; Y_S = -Y_E, \;\; Z_S = -Z_E$$

относительно центра Земли в небесной системе отсчёта.

Вычислить косинус угла H_1 между направлениями из центра Земли на Солнце и на пункт наблюдений

$$\cos H_1 = \frac{\bar{X}_q X_S + \bar{Y}_q Y_S + \bar{Z}_q Z_S}{\sqrt{\bar{X}_q^2 + \bar{Y}_q^2 + \bar{Z}_q^2}\,\sqrt{X_S^2 + Y_S^2 + Z_S^2}} .$$

Если $\cos H_1 > 0$, то на пункте наблюдений день, если $\cos H_1 < -0.17$, то на пункте наблюдений ночь, иначе – сумерки.

11.6 Видимая звёздная величина

Исходные данные содержат числовые значения двух параметров: абсолютную звёздную величину H и параметр наклона G. С помощью этих параметров можно получить оценку видимой звёздной величины v_b малого тела.

Пусть $x(t')$, $y(t')$, $z(t')$ координаты объекта в гелиоцентрической экваториальной системе отсчёта, а величины x_f, y_f, z_f – видимые геоцентрические координаты объекта.

Вычислим гелиоцентрическое расстояние r

$$r = \sqrt{x(t')^2 + y(t')^2 + z(t')^2}.$$

Вычислим геоцентрическое расстояние Δ

$$\Delta = \sqrt{x_f^2 + y_f^2 + z_f^2}.$$

Пусть β – угол фазы. Тогда

$$\begin{aligned}
\cos\beta &= \frac{x(t')\,x_f + y(t')\,y_f + z(t')\,z_f}{r\,\Delta}\,,\\
\sin\beta &= \sqrt{1-\cos^2\beta},\\
\operatorname{tg}\frac{\beta}{2} &= \frac{\sin\beta}{1+\cos\beta}\,.
\end{aligned}$$

Далее следует определить числовые значения двух функций угла фазы:

$$\begin{aligned}
\Phi_1 &= \mathrm{Exp}\left\{-3.33\cdot\left[\operatorname{tg}\frac{\beta}{2}\right]^{0.63}\right\},\\
\Phi_2 &= \mathrm{Exp}\left\{-1.87\cdot\left[\operatorname{tg}\frac{\beta}{2}\right]^{1.22}\right\}.
\end{aligned}$$

Оценка видимой звёздной величины вычисляется по формуле

$$v_b = H + 5\,\lg(r\Delta) - 2.5\,\lg\big[(1-G)\Phi_1 + G\Phi_2\big].$$

11.7 Азимут, высота, часовой угол и склонение

Азимут A, угол высоты h и дальность d до объекта вычисляются по формулам:

$$\begin{aligned}
\cos A &= +\frac{x_q}{\sqrt{x_q^2+y_q^2}},\\
\sin A &= -\frac{y_q}{\sqrt{x_q^2+y_q^2}},\\
\sin h &= \frac{z_q}{\sqrt{x_q^2+y_q^2+z_q^2}},\\
\cos h &= +\sqrt{1-\sin^2 h},\\
d &= \sqrt{x_q^2+y_q^2+z_q^2}.
\end{aligned}$$

Система управления телескопа с экваториальной монтировкой требует знания установочных параметров: склонения δ и часового угла t_h. Для этого надо выполнить следующие действия.

С помощью алгоритма разд.6.4 на с.26 вычислить значение истинного звёздного времени $S_\oplus$.

Топоцентрические координаты x_b, y_b, z_b следует перевести из земной системы координат в истинную экваториальную:

$$x_a = +x_b \cos S_\oplus - y_b \sin S_\oplus,$$
$$y_a = +x_b \sin S_\oplus + y_b \cos S_\oplus,$$
$$z_a = +z_b.$$

Прямое восхождение и склонение вычисляются по формулам

$$\begin{aligned} \cos\alpha &= \frac{x_a}{\sqrt{x_a^2 + y_a^2}}, \\ \sin\alpha &= \frac{y_a}{\sqrt{x_a^2 + y_a^2}}, \\ \sin\delta &= \frac{z_a}{\sqrt{x_a^2 + y_a^2 + z_a^2}}, \\ \cos\delta &= +\sqrt{1 - \sin^2\delta}. \end{aligned}$$

Часовой угол t_h, прямое восхождение α, истинное звёздное время $S_\oplus$ и долгота места наблюдения λ_p связаны формулой:

$$t_h = S_\oplus + \lambda_p - \alpha.$$

11.8 Контрольный пример

Использованы начальные значения оскулирующих кеплеровских элементов орбиты астероида Апофис (с.5). Вычисление видимой звёздной величины реализовано в модуле **unastvam.c**.

Прототип функции

```
double togetmau ( double h ,        /* absolute magnitude */
                  double g ,        /* slope parameter */
                  double p[4] ,     /* vector from the Sun */
                  double r[4] ) ;   /* vector from the Earth */
```

12 Результаты расчётов

В настоящее время астероид Апофис относительно наземного наблюдателя находится на солнечной стороне.

Очередной период ночной видимости состоится с января по апрель 2012 года. В январе 2013 года объект, имея отрицательное склонение, приближается к Земле на расстояние 13 миллионов километров. Ситуация повторяется с декабря 2019 года по март 2021 года и в начале 2028 года. За месяц до даты 13 апреля 2029 года – момента наибольшего сближения астероида с Землёй – условия его наблюдений в северном полушарии неблагоприятные. Целеуказания на несколько моментов времени для наблюдателя обсерватории на пике Терскол представлены в табл. 4.

Таблица 4: Условия видимости

дата	UTC	α (J2000)	δ (J2000)	δr (AU)	A (°)	h (°)	m
2012 01 16	17^h00^m	$+00^h20^m21\overset{s}{.}60$	$-02°20'49\overset{''}{.}1$	0.61347	$228\overset{\circ}{.}01$	$31\overset{\circ}{.}67$	20.6
2012 01 17	17 00	+00 25 17.05	−01 53 05.3	0.61408	228.05	32.19	20.6
2012 02 15	17 00	+02 37 32.93	+10 04 07.7	0.67545	231.64	44.61	20.8
2013 01 15	00 00	+08 38 13.62	−21 54 08.0	0.09771	198.83	21.46	15.9
2013 02 12	19 00	+06 52 09.17	−00 43 26.8	0.14386	178.27	44.85	16.8
2013 03 10	18 00	+06 50 43.30	+11 09 46.8	0.22007	197.06	55.71	18.2
2019 12 31	17 00	+23 26 17.43	−09 14 35.9	0.44915	221.77	26.92	20.2
2020 01 18	17 00	+01 02 13.70	−00 29 25.1	0.45968	220.15	37.55	20.1
2020 02 16	17 00	+03 12 45.52	+11 10 43.4	0.54368	222.91	49.70	20.4
2028 02 20	17 00	+04 01 40.53	+12 26 31.7	0.43585	211.90	54.46	19.9
2029 04 01	23 00	+14 12 33.92	−30 14 57.8	0.04106	176.43	15.17	13.7
2029 04 11	22 00	+14 05 22.38	−29 54 40.0	0.00692	173.39	15.29	9.5
2029 04 12	22 00	+13 55 03.11	−29 00 05.3	0.00350	176.54	16.41	8.0
2029 04 13	20 00	+11 19 02.62	−06 18 25.5	0.00035	187.85	38.84	3.3
2029 04 13	21 00	+09 44 09.28	+09 26 10.4	0.00025	241.78	37.79	3.4

В колонке α дано прямое восхождение объекта в часовой мере. В колонке δ содержатся значения склонения в градусной мере. В колонках A и h представлены, соответственно, числовые значения геодезического азимута и угловой высоты объекта над горизонтом.

Список литературы

[1] *Микиша А.М., Смирнов М.А., Барабанов С.И., Багров А.В., Болгова Г.Т., Рыхлова Л.В. Угроза с неба: рок или случайность?* М., Космосинформ, 1999.

[2] *Виноградова Т.А., Железнов Н.Б., Кузнецов В.Б., Чернетенко Ю.А., Шор В.А. Каталог потенциально опасных астероидов и комет.* /Труды ИПА РАН (эфемеридная астрономия), 2004, т.9.

[3] *Куимов К.В. Редукционные вычисления.* /В сб.: Практикум по астрометрии, изд.-во Московского университета, 1989, с.6-42.

[4] *Абалакин В.К. Основы эфемеридной астрономии.* М., Наука, 1979.

[5] *Абалакин В.К., Аксёнов Е.П., Гребеников Е.А., Дёмин В.Г., Рябов Ю.А. Справочное руководство по небесной механике и астродинамике. Под ред. Г.Н.Дубошина.* М., Наука, 1976.

[6] *Everhart E. Implicit single-sequence methods for integrating orbits.* //Celestial Mechanics, 1974, vol.10, p.35-55.

[7] *Бордовицына Т.В. Современные численные методы в задачах небесной механики.* М., Наука, 1984.

[8] *Татевян С.К., Сорокин Н.А., Залёткин С.Ф. Об одном методе численного интегрирования дифференциальных уравнений первого и второго порядка в астродинамике и космической геодезии.* /Пакеты прикладных программ. М., изд-во МГУ, 1997. С.60-119.

[9] *Монтенбрук О., Пфлегер Т. Астрономия на персональном компьютере.* С.-Пб., изд-во Питер, 2002.

[10] Яров-Яровой М.С. О применении уточнённых методов численного интегрирования в небесной механике. //Труды Государственного астрономического института им. П.К.Штернберга, 1974, т.45, с.178-200.