УДК 521.182.2

Численное определение траекторий ИСЗ методом Адамса переменного порядка. II

В. К. Тарадий, М. Л. Цесис

Предлагаются алгоритмы и программа численного интегрирования методом Адамса переменного порядка систем обыкновенных дифференциальных уравнений первого, второго и смешанных порядков.

NUMERICAL DETERMINATION OF THE ARTIFICIAL EARTH SATELLITE TRAJECTORIES BY THE ADAMS VARIABLE ORDER METHOD. II, by Taradij V. K., Tsesis M. L.— Algorithms and a programme are suggested for the Adams variable order numerical integration for systems of ordinary differential equations of the first, second and mixed orders.

При вычислениях положений и скоростей движения искусственных спутников Земли (ИСЗ) в ряде случаев возникает необходимость интегрировать системы обыкновенных дифференциальных уравнений смешанных (первого и второго) порядков (например, системы уравнений в параметрических переменных Кустаанхеймо-Штифеля [4]). В общем виде такие системы могут быть представлены соотношениями:

$$\ddot{x}_1 = f_1(t, \vec{x}, \dot{\vec{x}}),$$

$$\ddot{x}_2 = f_2(t, \vec{x}, \dot{\vec{x}}),$$

$$\cdots\cdots$$

$$\ddot{x}_n = f_n(t, \vec{x}, \dot{\vec{x}}),$$

$$\dot{x}_{n+1} = f_{n+1}(t, \vec{x}, \dot{\vec{x}}),$$

$$\dot{x}_{n+m} = f_{n+m}(t, \vec{x}, \dot{\vec{x}}),$$

где $\vec{x}$ — $n+m$-мерный вектор с компонентами: x_1, x_2, ... x_{n+m}, а t — независимая переменная. При построении алгоритма решения их методом Адамса нами были использованы формулы численного интегрирования, приведенные в [3] (с. 92) и в [2]. Алгоритм обеспечивает интегрирование систем уравнений первого, второго и смешанных порядков. При этом число уравнений в системе должно удовлетворять соотношению $0<2n+m<30$, где n и m — соответственно число уравнений второго и первого порядков. Для организации оптимального режима интегрирования в алгоритме предусмотрены следующие возможности. По желанию пользователя может быть осуществлен контроль за поведением либо относительной, либо абсолютной погрешностей (причем по выбору могут контролироваться погрешности решений или погрешности их первых производных). Траекторные вычисления могут производиться по одной из двух схем, а именно: PEC или PECE. Выбор длины шага и величины порядка алгоритма осуществляется индивидуально для каждого из уравнений системы. Ниже приводятся описания программных реализаций алгоритма интегрирования и контрольного примера. Тексты программ на языке ФОРТРАН даны в конце статьи.

Обращение к интегратору из вызывающего программного модуля имеет вид: CALL VASOMI (TNABL, YM), где TNABL — передаваемое пользователем в подпрограмму значение независимой переменной t; YM — массив, содержащий соответствующие TNABL значения зависимых переменных и их производных. Элементы массива располагаются в следующем порядке: $x_1, x_2, \ldots, x_n, \dot{x}_1, \dot{x}_2, \ldots, \dot{x}_n, x_{n+1}, x_{n+2}, \ldots, x_{n+m}$.

Имеются два основных режима работы интегратора, способы задания которых описываются ниже.

Режим 1. Пользователь передает в интегратор значение TNABL. В момент, когда независимая переменная t достигает в процессе численного интегрирования TNABL, определяются с помощью интерполяции элементы массива YM, после чего происходит возврат в вызывающий программный модуль. Обработав информацию, полученную в какой-то момент, пользователь может вновь обратиться к VASOMI, передав новое значение TNABL. Непрерывность процесса интегрирования при этом не нарушается. Последовательность значений TNABL должна монотонно изменяться в направлении интегрирования.

Режим 2. Возврат в вызывающий программный модуль происходит после завершения каждого шага интегрирования. При этом интерполяция на момент TNABL не производится, а само значение TNABL игнорируется. Режим 2 может быть использован при обработке наблюдений на основе систем регуляризированных уравнений.

Основной обмен информацией с программным модулем, вызывающим VASOMI, осуществляется с помощью следующего COMMON-блока:

COMMON | ANF | T, Y, H, EPS, NQ, MAXNQ, IAR, IO1,

*KKOR, KOTW, NV, NVR, INTER, IVOSW, JSTART,

где T — текущее значение независимой переменной t (начальное значение задается пользователем и изменяется автоматически); Y — массив, содержащий текущее значение зависимых переменных и их производных. Начальные значения (условия задачи Коши) задаются пользователем и располагаются в следующем порядке: $x_1, x_2, \ldots, x_n, \dot{x}_1, \dot{x}_2, \ldots$ $\ldots, \dot{x}_n, x_{n+1}, x_{n+2}, \ldots, x_{n+m}$; общее количество элементов последовательности равно $2n+m$; H — величина текущего шага интегрирования. Начальное значение задается пользователем. Знак начального значения длины шага определяется направлением интегрирования. В процессе интегрирования величина текущего шага изменяется автоматически. EPS — массив, содержащий $n+m$ величин, предназначенных контролировать погрешности на шаге интегрирования каждого из уравнений системы. Различным элементам массива могут присваиваться различные значения, с целью обеспечения оптимальной длины шага интегрирования. Массив задается пользователем перед первым обращением к VASOMI; NQ — массив, содержащий текущие значения порядка метода для каждого уравнения системы. Перед первым обращением к VASOMI необходимо всем элементам массива присвоить значение «1». Общее количество элементов — $n+m$; MAXNQ — массив, элементы которого содержат максимально допустимые величины порядка алгоритма для каждого из уравнений системы. Задается пользователем перед первым обращением к VASOMI. При вычислениях траекторий ИСЗ значения элементов массива MAXNQ, по-видимому, следует ограничивать пределами 14÷18; IAR — массив, каждому элементу которого пользователь должен присвоить значение «0» или «1». Если некоторому элементу IAR присвоено значение «0», то в процессе интегрирования соответствующего уравнения системы контролируется поведение абсолютной погрешности. В альтернативном варианте ($IAR_x=1$) осуществляется контроль над относительной погрешностью; IO1 — массив, каждому элементу которого пользователь должен присвоить значение «0» или «1». Если i-й элемент массива IO1 равен «0», то в процессе интегрирования вычисляется оценка погрешности (абсолютной или относительной) $\dot{x}_i$; в альтернативном варианте ($IO1_i=1$) оценивается погрешность x_i. Ясно, что элементам массива под номерами $n+1$, $n+2, \ldots$ $\ldots, n+m$ должны быть присвоены значения «0»; KKOR — переменная, которой пользователь присваивает значение «1» или «2» в зависимости от того, какой алгоритм решения предполагается использовать (PEC или PECE соответственно); KOTW — число отвергнутых шагов, т. е. число шагов, на которых требуемая точность интегрирования не была достигнута и поэтому вычисления пришлось повторять с уменьшенной величиной шага. Перед первым обращением к VASOMI пользователь должен присвоить KOTW значение «0». NV — переменная, которой пользователь присваивает значение $2n+m$; NVR — переменная, которой пользователь присваивает значение $n+m$; INTER — переменная, которой пользователь присваивает значение «0» или «1». Если INTER$=1$, то подпрограмма осуществляет интерполяцию решения во внесеточные точки, т. е. в точки, расположенные внутри шага интегрирования. В противном случае интерполяция отсутствует, и значение TNABL игнорируется; IVOSW — переменная, которой пользователь присваивает значение «0» или «1». Если IVOSW$=1$, то возврат в вызывающий VASOMI программный модуль происходит после завершения каждого шага интегрирования. При этом содержание YM может быть неопределенным. В случае, когда IVOSW$=0$, возврат происходит только после достижения TNABL и определения элементов массива YM (если INTER$=1$); JSTART — переменная, содержащая значение текущего максимального порядка алгоритма интегрирования (определяется путем сравнения друг с другом элементов массива NQ). Перед первым обращением к VASOMI пользователь должен присвоить JSTART значение «0».

Подпрограмма вычисления правых частей составляется пользователем и должна иметь заголовок:

SUBROUTINE DIFFUN (N, T, Y, F),

где N — переменная, значение которой равно $2n+m$; T — независимая переменная t; Y — массив зависимых переменных, соответствующих T и расположенных в следующем порядке: $x_1, x_2, \ldots, x_n, \dot{x}_1, \dot{x}_2, \ldots, \dot{x}_n, x_{n+1}$, $x_{n+2}, \ldots, x_{n+m}$; F — массив правых частей уравнений. Значения функций $f_1, f_2, \ldots, f_{n+m}$ размещаются в этом массиве последовательно, начиная с элемента под номером $n+1$. Первые n элементов могут принимать произвольные значения. Такое расположение принято для удобства приведения (по желанию пользователя) интегрируемой системы к эквивалентной системе уравнений первого порядка. В этом случае изменения в подпрограмме DIFFUN ограничиваются присвоением первым n элементам массива F последовательно значений: $\dot{x}_1, \dot{x}_2, \ldots, \dot{x}_n$, соответствующих значению T.

Расположенный в подпрограмме VASOMI COMMON-блок COMMON|NULE|NULE содержит единственную переменную NULE, используемую в качестве индикатора того, что пользователь затребовал слишком высокую точность на шаге интегрирования. Если на некотором шаге оценка погрешности какой-либо переменной становится равной машинному нулю, к прежнему значению NULE автоматически прибавляется единица. Перед первым обращением к VASOMI необходимо задать NULE$=0$. Если в процессе интегрирования оказывается, что NULE$\neq 0$, значит некоторые из элементов массива EPS являются слишком малыми величинами для данной ЭВМ. Это ведет к ухудшению результатов интегрирования.

Примечание. Все рассмотренные выше массивы имеют размерность 30 и могут быть использованы для решения системы уравнений, где $0<2n+m<30$.

Для конкретных значений n и m массивы, содержащие значения зависимых переменных и их производных, а также массив F из подпрограммы DIFFUN, используют $2n+m$ элементов, начиная с первого. Все остальные массивы используют $n+m$ начальных элементов. Элементы, оставшиеся неопределенными, игнорируются программой.

В контрольном примере [1] приводятся результаты интегрирования задачи 2-х тел в прямоугольных координатах. С целью обеспечения большей эффективности тестирования одно из трех уравнений 2-го порядка заменено двумя уравнениями 1-го порядка. В итоге непосредственно решаемая система уравнений имеет вид:

$$\ddot{x}_1 = -\frac{\mu x_1}{r^3}, \quad \ddot{x}_2 = -\frac{\mu x_2}{r^3},$$

$$\dot{x}_3 = x_4, \quad \dot{x}_4 = -\frac{\mu x_3}{r^3},$$

где $r=\sqrt{x_1^2+x_2^2+x_3^2}$, $\mu = 11467.8239$ э.р.^{3}/сут^2.

Для данной системы ($n=2$, $m=2$) начальные условия задачи Коши таковы: $t_0=0$, $x_1=0.59$ э. р., $x_2=0.91$ э. р., $\dot{x}_1=-115.19$ э. р./сут, $x_2=58.23$ э. р./сут, $x_3=0.26$ э. р., $x_4=57.44$ э. р./сут.

На печать выводятся результаты интегрирования, полученные в момент TNABL=5 сут:

—0.8862737258122277 01—0.4455440053576726 01—0.2794474025113324 02

—0.2975206274560535 02 0.3207958296499912 00—0.6191590957581975 01

В первой строке расположены: x_1, x_2, $\dot{x}_1$, а во второй—$\dot{x}_2$, x_3, $\dot{x}_3$.

Текст программы VASOMI

```
      SUBROUTINE VASOMI(TNABL, YM)
      IMPLICIT REAL*8(A—H, O—Z)
      REAL*8 LY1, LY2
      REAL SEM, BER
      INTEGER*2 K0(4)
      COMMON/NULE/NULE
      COMMON/ANF/T, Y, HH, EPS, NQ, MAFNQ, IAR, I01,
     *KKOR, KOTW, NV, NVR, INTER, IVOSW, JSTART
      DIMENSION YM(30), Y(30), EPS(30), NQ(30), MAXNQ(30)
      DIMENSION IAR(30), I01(30)
      DIMENSION YS(30), YP(30), YMAX(30), FP(30), FC(30)
      DIMENSION FPE(30), FPAR(30)
      DIMENSION ER(30), BND(30)
      DIMENSION TR(30), G(30, 30), S(30, 30), RA(30, 30)
      EQUIVALENCE (K0(1), H)
      KUDA=1
      IF(HH. LT. 0D0) KUDA=—1
    1 CONTINUE
      IF(INTER. EQ. 1. AND. JSTART. EQ. —1) GO TO 407
      LASA=NV
      H=HH
      NV2=NV—NVR
      DO 3 I=1. NVR
```

```
      J=I+NV2*(1—I01(I))
      YMAX(I)=DMAX1 (DABS(Y(J)). 1D—13)
      IF(IAR(I). EQ. 0) YMAX(I)=1
    3 BND(I)=(YMAX(I)*EPS(I))/(6D0*DFLOAT(NQ(I)+1))
      IF(JSTART. EQ. 100) KEQH=1
      IF(JSTART. EQ. 100) GO TO 61
      K0(2)=0
      K0(3)=0
      K0(4)=0
   61 CONTINUE
      DO 47 I=1, NV
   47 YS(I)=Y(I)
      TOLD=T
      T=T+H
      NH=0
      DO 4 I=1, NVR
      NH=MAX0(NH, NQ(I))
    4 CONTINUE
      NH1=NH+1
      NH2=NH+2
      NH3=NH+3
      IF(JSTART. NE. 0) GO TO 300
   48 KEQH=1
      G(1, 1)=1D0
      G(1, 2)=H
      G(1, 3)=H*H/2D0
      G(1, 4)=H*H*H/6D0
      G(2, 1)=H
      G(2, 2)=G(1, 3)
      G(2, 3)=G(1, 4)
      TR(1)=H
      TR(2)=H
      CALL DIFFUN (LASA, TOLD, YS, FPAR)
      DO 2 I=1, NVR
      NQ(I)=1
    2 RA(I, 1)=FPAP (I+NV2)
      NHOLD=1
      HOLD=H
      GO TO 301
  300 PROM=H
      PROM1=TR(1)
      DO 502 I=1, NH2
      TR(I)=PROM
      PROM=PROM1
  502 PROM1=TR(I+1)
      IF(KEQH. GT.1) GO TO 503
  302 G(1, 1)=1D0
      DO 500 I=1, NH3
  500 G(1, I+1)=G(1, I)*H/DFLOAT(I)
      SUMH=0
      DO 501 I=2, NH2
      I1=NH3—I+2
      SUMH=SUMH+TR(I—1)
      DO 501 J=1, I1
  501 G(I, J)=SUMH*G(I—1, J)—DFLOAT(J—1)*G(I—1, J+1)
      GO TO 301
  503 IF(NH. GT. NHOLD) GO TO 505
      IF(KEQH. GE. NH2) GO TO 301
      SUMH=H*DFLOAT (KEQH—1)
```

```
      DO 504 I=KEQH1, NH2
      I1=NH3—I+2
      SUMH=SUMH+TR(I—1)
      DO 504 J=1, I1
504   G(I, J)=SUMH*G(I — 1, J) — DFLOAT(J — 1)*G(I — 1, J+1)
      GO TO 301
505   I2=MINO (KEQH, NH2)
      G(1, NH+4)=G(1, NH3)*H/DFLOAT(NH3)
      SUMH=0
      DO 506 I=2, I2
      J=NH3—I+2
      SUMH=SUMH+TR(I—1)
506   G(I, J)=SUMH*G(I — 1, J) — DFLOAT(J — 1)*G(I — 1, J+1)
      IF (KEQH. GE. NH2) GO TO 301
      SUMH=H*DFLOAT(KEQH—1)
      DO 507 I=KEQH1, NH2
      I1=NH3—I+2
      SUMH=SUMH+TR(I—1)
      DO 507 J=1, I1
507   G(I, J)=SUMH*G(I—1, J)—DFLOAT(J—1)*G(I—1, J+1)
301   DO 400 I=1, NVR
      NA=NQ(I)
      LY1=0
      LY2=0
      DO 40 J=1, NA
      JI=NA—J+1
      LY1=LY1+G(JI, 2)*RA(I, JI)
      IF (I. GT. NV2) GO TO 40
      LY2=LY2+G(JI, 3)*RA(I, JI)
40    CONTINUE
      YP (I+NV2)=LY1+YS(I+NV2)
      IF(I. GT. NV2) GO TO 400
      YP(I)=LY2+H*YS(I+NV2)+YS(I)
400   CONTINUE
      CALL DIFFUN (LASA, T, YP, FPAR)
      DO 402 I=1, NVR
402   FP(I)=FPAR(I+NV2)
      DO 42 I=1, NVR
      NA=NQ(I)
      FSW=0
      DO 41 J=1, NA
      JI=NA—J+1
41    FSW=FSW+G(JI, 1)*RA(I, JI)
      FPE(I)=FP(I)—FSW
      G10=G(NA+1, 2)/G(NA+1, 1)
      IF(NA. EQ. 1) G10=H/2D0
      Y(I+NV2)=YP(I+NV2)+G10*FPE(I)
      IF(I. GT. NV2) GO TO 42
      G20=G(NA+1,3)/G(NA+1, 1)
      IF(NA. EQ. 1) G20=H*H/6D0
      Y(I)=YP(I)+G20*FPE(I)
42    CONTINUE
      NT=NVR
      DO 421 I=1, NVR
      NA=NQ(I)
      J=I01(I)
      A0=DFLOAT(1+J)*(G(NA, 3+J)/G(NA+1, 1))
      ER(I)=DABS(A0*FPE(I))
      IF(ER(I). LT. BND(I)) NT=NT—1
```

```
  421 CONTINUE
      IF(NT. LE. 0) GO TO 422
      H=H/2D0
      K0(2)=0
      K0(3)=0
      K0(4)=0
      TR(1)=H
      KOTW=KOTW+1
      KEQH=1
      T=TOLD+H
      IF(JSTART. EQ. 0) GO TO 48
      GO TO 302
  422 CONTINUE
      IF(KKOR. EQ. 1) GO TO 412
      CALL DIFFUN (LASA, T, Y, FPAR)
  412 CONTINUE
      DO 413 I=1. NVR
  413 FC(I)=FPAR(I+NV2)
  414 CONTINUE
  407 CONTINUE
      IF(INTER. NE. 1. OR. (T—TNABL)*KUDA. LT. 0D0) GO TO 408
      HXP=TR(1)
      H=TNABL—TOLD
      TR(1)=H
      S(1, 1)=1D0
      DO 520 I=1, NH3
  520 S(1, I+1)=S(1,I)*H/DFLOAT(I)
      SUMH=0
      DO 521 I=2, NH2
      I1=NH3—I+2
      SUMH=SUMH+TR(I—1)
      DO 521 J=1, I1
  521 S(I, J)=SUMH*S(I—1, J)—DFLOAT(J—1)*S(I—1, J+1)
      DO 405 I=1, NVR
      NA=NQ(I)
      LY1=0
      LY2=0
      DO 406 J=1, NA
      JI=NA—J+1
      LY1=LY1+S(JI, 2)*RA(I, JI)
      IF(I. GT. NV2) GO TO 406
      LY2=LY2+S(JI, 3)*RA(I, JI)
  406 CONTINUE
      YM(I+NV2)=LY1+YS(I+NV2)
      IF(I. GT. NV2) GO TO 405
      YM(I)=LY2+H*YS(I+NV2)+YS(I)
  405 CONTINUE
      H=HXP
      TR(1)=HXP
      JSTART=—1
      RETURN
  408 CONTINUE
      IF(JSTART. EQ. 0) GO TO 59
      SEM=1E+20
      DO 418 I=1, NVR
      NA=NQ(I)
      J=I01(I)
      AM=DFLOAT(1+J)*(G(NA—1,3+J)/G(NA, 1))
      AP=DFLOAT (1+J)*(G(NA+1,3+J)/G(NA+2,1))
```

```
      ERM1=DABS (AM*(FPE(I)+G(NA, 1)*RA(I, NA)))
      IF(NA. EQ. 1) ERM1=1D+20
      ER0=ER(I)
      ERP1=DABS(AP*(FPE(I)—G(NA+1,1)*RA(I, NA+1)))
      IF(NA. EQ. MAXNQ(I)) ERP1=1D+20
      ERN=DMIN1 (ERM1, ER0, ERP1)
      IF(ERN. EQ. ERM1)NA=NA—1
      IF(ERN. EQ. ERP1)NA=NA+1
      BER=2
      IF(ERN. NE. 0D0) BER=BND(I)/ERN
      IF(ERN. NE. 0D0) BER=BER**(1E0/FLOAT(NA+1+J))
      IF(ERN. EQ. 0D0) NULE=NULE+1
      IF(SEM. GE. BER) SEM=BER
      NQ(I)=NA
  418 CONTINUE
      HNEW=H*SEM
  451 HOLD=H
      NHOLD=NH
      H=HNEW
      K0(2)=0
      K0(3)=0
      K0(4)=0
      IF(DABS((H—HOLD)/H). LT. 0. 1D0)H=HOLD
      IF(H. EQ. HOLD) KEQH=KEQH+1
      IF(H. NE. HOLD) KEQH=1
      KEQH1=KEQH+1
   59 DO 60 I=1, NVR
      PP0=RA(I, 1)
      RA(I, 1)=FC(I)
      SUMH=0
      NA=NQ(I)
      DO 60 J=1, NA
      SUMH=SUMH+TR(J)
      PP1=(RA(I, J)—PP0)/SUMH
      PP0=RA(I, J+1)
   60 RA(I, J+1)=PP1
      JSTART=NH
      HH=H
      IF(IVOSW. EQ. 1) RETURN
      GO TO 1
      END
```

Текст программы контрольного примера

```
      IMPLICIT REAL*8(A—H, O—Z)
      COMMON/ANF/T, Y, H, EPS, NQ, MAXNQ, IAR, I01,
     *KKOR, KOTW, NV, NVR, INTER, IVOSW, JSTART
      COMMON/NULE/NULE
      COMMON/KW/KOBR
      DIMENSION Y(30), EPS(30), NQ(30), MAXNQ(30)
      DIMENSION IAR(30), I01(30), YM(30)
      CALL NUNDFL
      X0=0.59D0
      Y0=0.91D0
      Z0=0.26D0
      VX0=—115. 19D0
      VY0=58. 23D0
      VZ0=57. 44D0
      NV=6
```

```
      NVR=4
      Y(1)=X0
      Y(2)=Y0
      Y(3)=VX0
      Y(4)=VY0
      Y(5)=Z0
        Y(6)=VZO
      DO 1 I=1. NVR
      EPS(I)=1D—10
      NQ(I)=1
      MAXNQ(I)=15
      IAR(I)=0
      I01(I)=0
      1 CONTINUE
      T=0
      H=0. 1D0
        KOBR=0
        KKOR=2
      KOTW=0
      NULE=0
      JSTART=0
      INTER=1
      IVOSW=0
      TNABL=5
      CALL VASOMI (TNABL. YM)
      WRITE (3,2) (YM(I), I=1, NV)
      2 FORMAT(' ', 3D30, 16)
      STOP
      END
      SUBROUTINE DIFFUN (N, X, Y, F)
      IMPLICIT REAL*8(A—H, O—Z)
      COMMON/KW/KWI
      DIMENSION F(1), Y(1)
        KWI=KWI+1
      R=DSQRT(Y(1)*Y(1)+Y(2)*Y(2)+Y(5)*Y(5))
      R3=R*R*R
      FM=11467.8239D0
      F(3)=—FM*Y(1)/R3
      F(4)=—FM*Y(2)/R3
      F(5)=Y(6)
      F(6)=—FM*Y(5)/R3
      RETURN
      END
```


1. *Макарова Е. Н.* О численном интегрировании уравнений движения ИСЗ в случае орбит с большим эксцентриситетом.— Бюл. Ин-та теор. астрон. АН СССР, 1979, **14**, с. 486—489.
2. *Тарадий В. К., Цесис М. Л.* Численное определение траекторий ИСЗ методом Адамса переменного порядка. I.— Астрометрия и астрофизика, 1984, вып. 53, с. 56—65.
3. *Хол Дж., Уатт Дж.* Современные численные методы решения обыкновенных дифференциальных уравнений.— М.: Мир, 1979.— 312 с.
4. *Штифель Е., Шейфеле Г.* Линейная и регулярная небесная механика.— М.: Наука, 1975.— 303 с.

Главная астрономическая обсерватория АН УССР,
Киев

Поступила в редакцию
16.07.1984