Initial value problem in ordinary differential equations

Subroutine RKF45

The sample main program at the end of this section uses RKF45 to find the motion of two bodies under mutual gravitational attraction. Let $x(t)$ and $y(t)$ denote the position of one body in a coordinate system with the origin fixed in the other body. The differential equations derived from Newton's laws of motion are

$$x''(t) = \frac{-\alpha^2 x(t)}{R(t)}$$

$$y''(t) = \frac{-\alpha^2 y(t)}{R(t)},$$

where

$$R(t) = [x(t)^2 + y(t)^2]^{3/2}$$

and α is a constant involving the gravitational constant, the masses of the two bodies, and the units of measurement.

If the initial conditions are chosen as

$$x(0) = 1 - e, \qquad x'(0) = 0,$$

$$y(0) = 0, \qquad y'(0) = \alpha\left(\frac{1+e}{1-e}\right)^{1/2},$$

for some parameter e with $0 \leq e < 1$, then the solution turns out to be periodic with period $2\pi/\alpha$. The orbit is an ellipse with eccentricity e and with one focus at the origin.

To write this as a system of four first-order equations, we introduce

$$y_1 = x, \qquad y_2 = y, \qquad y_3 = x', \qquad y_4 = y'.$$

The equations and initial conditions then become

$$R = \frac{(y_1^2 + y_2^2)^{3/2}}{\alpha^2},$$

$$y_1' = y_3, \qquad y_1(0) = 1 - e,$$

$$y_2' = y_4, \qquad y_2(0) = 0,$$

$$y_3' = -\frac{y_1}{R}, \qquad y_3(0) = 0,$$

$$y_4' = -\frac{y_2}{R}, \qquad y_4(0) = \alpha\left(\frac{1+e}{1-e}\right)^{1/2}.$$

By rescaling the time variable, it is possible to eliminate α, but we have not done this because we wish to illustrate the use of Fortran Common to

pass parameters such as α from the main program to the subroutine defining the equations.

The parameter IFLAG is an important control variable. It should be set to 1 for the first entry to RKF45. Ordinarily, RKF45 will reset it to 2, and it should be left at 2 for subsequent entries. Values other than 2 returned by RKF45 signal various warning and error conditions described in detail in the comments. IFLAG = 4 and IFLAG = 7 are warnings that RKF45 must work very hard to obtain the requested accuracy. It is possible to continue, but the user may want to consider increasing the error tolerances or changing to a subroutine which uses a multistep method. IFLAG = 3 indicates that too much relative accuracy is being requested, and IFLAG = 5 or 6 indicates that the error tolerances must be changed before continuing. IFLAG = 8 indicates that RKF45 is being called incorrectly. The user is strongly advised to include a check on IFLAG in his main program.

In this sample run, we have taken $e = 0.25$ and $\alpha = \pi/4$ and have printed the position for $0 \leq t \leq 12$ in steps of 0.5. The output is in Table 6.2. Notice that the orbit is periodic with a period of $t = 8$.

Table 6.2 OUTPUT FROM SAMPLE PROGRAM

0.0	0.750000000	0.000000000
0.5	0.619768032	0.477791373
1.0	0.294417538	0.812178519
1.5	−0.105176382	0.958038092
2.0	−0.490299793	0.939874996
2.5	−0.813942832	0.799590802
3.0	−1.054031517	0.575706078
3.5	−1.200735042	0.300160708
4.0	−1.250000001	−0.000000001
4.5	−1.200735042	−0.300160709
5.0	−1.054031517	−0.575706079
5.5	−0.813942832	−0.799590803
6.0	−0.490299793	−0.939874996
6.5	−0.105176383	−0.958038092
7.0	0.294417537	−0.812178518
7.5	0.619768031	−0.477791370
8.0	0.749999996	0.000000006
8.5	0.619768024	0.477791379
9.0	0.294417526	0.812178522
9.5	−0.105176395	0.958038091
10.0	−0.490299806	0.939874991
10.5	−0.813942843	0.799590794
11.0	−1.054031524	0.575706068
11.5	−1.200735047	0.300160697
12.0	−1.250000002	−0.000000011

```
C  SAMPLE PROGRAM FOR RKF45
C
      SUBROUTINE ORBIT (T, Y, YP)
      REAL T, Y(4), YP(4), R, ALFASQ
      COMMON ALFASQ
      R = Y(1)*Y(1) + Y(2)*Y(2)
      R = R*SQRT(R)/ALFASQ
      YP(1) = Y(3)
      YP(2) = Y(4)
      YP(3) = -Y(1)/R
      YP(4) = -Y(2)/R
      RETURN
      END
C
      EXTERNAL ORBIT
      REAL T,Y(4),TOUT,RELERR,ABSERR
      REAL TFINAL,TPRINT,ECC,ALFA,ALFASQ,WORK(27)
      INTEGER IWORK(5), IFLAG, NEQN
      COMMON ALFASQ
      ECC = 0.25
      ALFA = 3.141592653589/4.0
      ALFASQ = ALFA*ALFA
      NEQN = 4
      T = 0.0
      Y(1) = 1.0 - ECC
      Y(2) = 0.0
      Y(3) = 0.0
      Y(4) = ALFA*SQRT( (1.0 + ECC)/(1.0 - ECC) )
      RELERR = 1.0E-9
      ABSERR = 0.0
      TFINAL = 12.0
      TPRINT = 0.5
      IFLAG = 1
      TOUT = T
   10 CALL RKF45(ORBIT,NEQN,Y,T,TOUT,RELERR,ABSERR,IFLAG,WORK,IWORK)
      WRITE(6,11) T, Y(1), Y(2)
      GO TO (80,20,30,40,50,60,70,80), IFLAG
   20 TOUT = T + TPRINT
      IF (T .LT. TFINAL) GO TO 10
      STOP
   30 WRITE(6,31) RELERR,ABSERR
      GO TO 10
   40 WRITE(6,41)
      GO TO 10
   50 ABSERR = 1.0E-9
      WRITE(6,31) RELERR,ABSERR
      GO TO 10
   60 RELERR = 10.0*RELERR
      WRITE(6,31) RELERR,ABSERR
      IFLAG = 2
      GO TO 10
   70 WRITE(6,71)
      IFLAG = 2
      GO TO 10
   80 WRITE(6,81)
      STOP
C
   11 FORMAT(F5.1, 2F15.9)
   31 FORMAT(17H TOLERANCES RESET, 2E12.3)
   41 FORMAT(11H MANY STEPS)
   71 FORMAT(12H MUCH OUTPUT)
   81 FORMAT(14H IMPROPER CALL)
      END
```