Assignment 4–Integrate a linear system of ODE using Taylor

Yixie Shao

March 27, 2015

1 Linear system

The linear system is expressed as:

$$\dot{x}_1 = a \times x_1 + b \times x_2$$
$$\dot{x}_2 = c \times x_1 + d \times x_2$$

The system can be written as:

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = A \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

Here,

$$A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

Different types of eigenvalues of A will give different phase portraits.

2 center

Figure 1 uses $main\_ls\_flow$. The initial condition is: (1,1), (2,2), (3,3). The eigenvalues of matrix A is: $\pm i$.

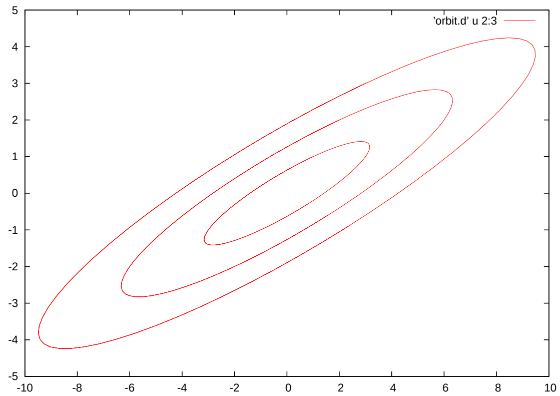

Figure 1: $a = 2, b = -5, c = 1, d = -2$

3 focus

Figure 2 uses *main_ls_flow*. The initial condition is: (1,0), (2,0), (3,0). ti=0, tmax=10, np=100.
The eigenvalues of matrix A is: $1 \pm 2i$.

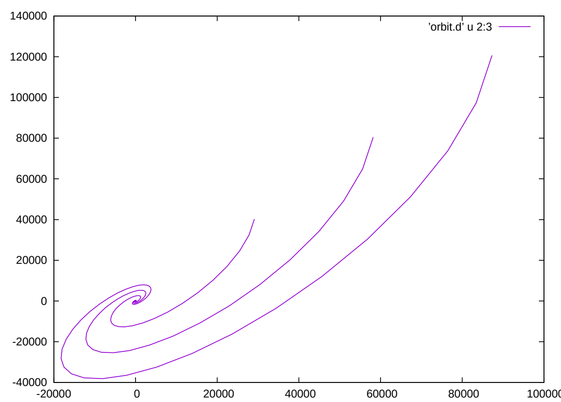

Figure 2: $a = 3, b = -2, c = 4, d = -1$

Figure 3 uses *main_ls_flow*. The initial condition is: (1,1), (2,5), (4,3). ti=0, tmax=10, np=100.

Figure 4 uses *main_ls_sec*1. The initial condition is: (1,0), (2,0), (3,0); idir=-1, n-crossing=4.

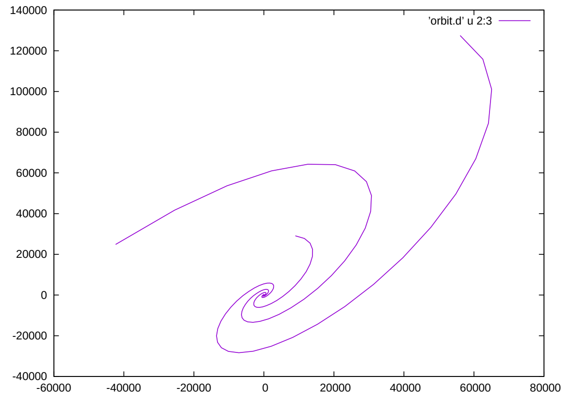

Figure 3: $a = 3, b = -2, c = 4, d = -1$

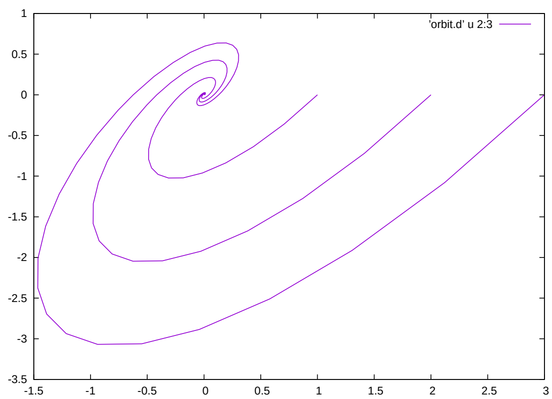

Figure 4: $a = 3, b = -2, c = 4, d = -1$

4 saddle

Figure 5 uses *main_ls_saddle*. The initial condition is: $(\pm 10^{-4}, \pm 3 \times 10^{-2})$, $(\pm 2 \times 10^{-4}, \pm 3 \times 10^{-2})$, $(\pm 4 \times 10^{-4}, \pm 3 \times 10^{-2})$. idir=-1, time period: 6.28, np=100.
From this data set and Figure 5, we can get that the vertical line is the unstable manifold (W^u, corresponding to eigenvalue: 2), and the other line the stable manifold (W^s, corresponding to eigenvalue: -1) of this linear system. And the point on the W^u has a higher speed than the one on W^s.

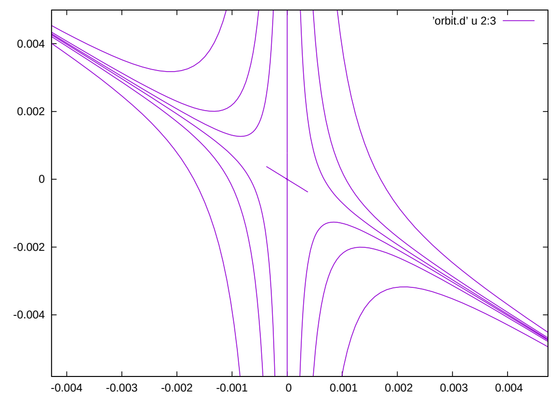

Figure 5: $a = -1, b = 0, c = 3, d = 2$

5 code

main_ls_flow

```
c
    ****************************************************************

c   MAIN_LS_FLOW.f

c   input: xi,ti,tmax,idir,np
c
    ****************************************************************

         implicit real*8 (a-h,o-z)
         parameter (n=2)
         common/param/aa,bb,cc,dd
         dimension xi(n),x(n)
         open(10,file='orbit.d',status='unknown')
         write(*,*) 'a,b,c,d'
         read(*,*) aa,bb,cc,dd
         write(*,*) 'ti,tmax,np (number of points)'
         read(*,*)ti,tmax,np
         do 23 j=1,3
         write(*,*) 'Initial condition x(1),x(2)'
         read(*,*) (xi(i),i=1,n)
         if (tmax.ge.ti)then
             idir=1
         else
             idir=-1
         endif
```

```
        do i=1,n
         x(i)=xi(i)
        enddo
         write(*,*)ti,'   initial t, initial cond:'
         write(*,*)(x(i),i=1,n)
       xinctime=dabs(tmax-ti)/np
         write (10,*)ti,(x(ii),ii=1,n)
       do 20 i=1,np
         call flow(ti,n,x,idir,xinctime)
         write (10,*)ti,(x(ii),ii=1,n)
 20    continue
       write(10,90)
 90    format()
       write(*,*)ti,'  final t,  final point:'
       write(*,*)(x(i),i=1,n)
 23    continue
        end

       subroutine flow(t,n,x,idir,xinctemps)
       IMPLICIT REAL*8 (A-H,O-Z)
       dimension x(n)
       tmax=t+idir*xinctemps
        hab=0.1e-16
        hre=0.1e-16
        pabs=dlog10(hab)
        prel=dlog10(hre)
        istep=1
        ht=0.d0
 1        CALL taylor_f77_eq_ls_(t,x,idir,istep,pabs,prel,
     &  tmax,ht,iordre,ifl)
        if (idir.eq.1.and.t.lt.tmax)go to 1
        if (idir.eq.-1.and.t.gt.tmax)go to 1
        if (dabs(t-tmax).le.1.d-13)return
        write(*,*)'problems in taylor'
        stop
        return
        end
```

main_ls_sec1

```
c
c  MAIN_LS_SEC1.f
c
c
   *****************************************************************

c
        implicit real*8 (a-h,o-z)
        parameter (n=2)
```

```
      common/param/aa,bb,cc,dd
      dimension yf(n),x(n)
      open(10,file='orbit.d',status='unknown')


      write(*,*) 'a,b,c,d'
      read(*,*) aa,bb,cc,dd
      write(*,*) 'idir?'
      read(*,*) idir
      write(*,*) 'm times crossing'
      read(*,*) m
      do 23 j=1,3
      write(*,*) 'Initial condition x(1),...,x(n)'
      read(*,*) (x(i),i=1,n)

      t=0.d0
      do 12 k=1,m
        write(10,*)t,(x(i),i=1,2)
        call poinc1(n,x,yf,tfinal,idir)
        x=yf
        t=tfinal+t
        write(*,*)t,'t'
 12   continue
      write(10,90)
 90   format()
 23   continue
      end


C
    ******************************************************************************

c Input:
c n dimension of the vectors yi and yf
c yi initial point
c idirorig: +1 integration forwards in time; -1 backwards
c yf final point
c tfinal final time
c
C
    ******************************************************************************

      SUBROUTINE POINC1(n,YI,YF,tfinal,idirorig)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION YI(n),YF(n),DGG(n),F(n)
      icont=0
      idir=idirorig
      ti=0.D0
```

```fortran
C      DETERMINATION OF THE FIRST PASSAGE OF THE ORBIT
    THROUGH y=0
         CALL SECCIO(YI,GG,DGG)
         IF(DABS(GG).LT.1.D-9)GG=0.d0
         GA=GG
         hab=.1e-16
         hre=.1e-16
         pabs=dlog10(hab)
         prel=dlog10(hre)
         istep=1
         pas=0.1d0
         ht=0.d0
         t=ti
 1         tmax=t+idir*pas
         CALL taylor_f77_eq_ls_(t,yi,idir,istep,pabs,prel,
     &   tmax,ht,iordre,ifl)
         CALL SECCIO(YI,GG,DGG)
         IF(GG*GA.LT.0.D0)go to 22
         write(10,*)t,(yi(ii),ii=1,2)
         GA=GG
         GO TO 1
 22       continue
         icont=icont+1
         if (icont.gt.20)then
             write(*,*)'problems finding the section'
             stop
         endif
         CALL FIELD(T,YI,N,F)
         P=0.D0
         DO 3 I=1,N
 3        P=P+F(I)*DGG(I)
         H=-GG/P
         if (h.ge.0.d0)idir=1
         if (h.lt.0.d0)idir=-1
         tmax=t+h
c           write(*,*)icont,'  refining: h and time ',h,tmax
c           write(*,*)'refining  t point ',t,yi(1),yi(2)
         CALL taylor_f77_eq_ls_(t,yi,idir,istep,pabs,prel,
     &   tmax,ht,iordre,ifl)
         CALL SECCIO(YI,GG,DGG)
         IF(DABS(GG).GT.1.D-13)GO TO 22
         DO 4 I=1,N
 4        YF(I)=YI(I)
         tfinal=t
         write(*,*)'tfinal point time ',tfinal
         write(*,*)(yf(ii),ii=1,n)
         write(10,*)t,(yf(ii),ii=1,2)
         return
         end
```

```
C
   ****************************************************************************
C
   *
C     THE SURFACE g OF SECTION,IN THIS CASE
C          INPUT PARAMETERS:
C     Y(*)        POINT
C          OUTPUT PARAMETERS:
C     GG           FUNCTION THAT EQUATED TO 0 GIVES THE
   SURFACE OF
C                  SECTION
C     DGG(*)       GRADIENT OF FUNCTION GG
C
   *
C
   ****************************************************************************
        SUBROUTINE SECCIO(Y,GG,DGG)
        IMPLICIT REAL*8(A-H,O-Z)
        DIMENSION Y(2),DGG(2)
        GG=Y(2)
        DO 1 I=1,2
 1      DGG(I)=0.D0
        DGG(2)=1.d0
        RETURN
        END


C
C FIELD.F
C
C
   ****************************************************************************
C
C     EQS OF MOTION IN synodical VARIABLES
C     X           TIME
C     Y(*)        POINT (Y(1),Y(2),....Y(n))
C     NEQ         NUMBER OF EQUATIONS
C          OUTPUT PARAMETERS:
C     F(*)        VECTOR FIELD
C
C
   ****************************************************************************
```

```
      subroutine field(t,x,neq,f)
      implicit real*8 (a-h,o-z)
      common/param/aa,bb,cc,dd
      dimension x(neq),f(neq)

      f(1) =aa*x(1)+bb*x(2)
      f(2)= cc*x(1)+dd*x(2)
      return
      end
```

main_ls_saddle

```
c
   ****************************************************************
c
c  MAIN_LS_SADDLE.f

c  input: xi,ti,tmax,idir,np
c
   ****************************************************************
        implicit real*8 (a-h,o-z)
        parameter (n=2)
        common/param/aa,bb,cc,dd
        dimension xi(n),x(n),a(n,n),rr(n),ri(n),vr(n,n),vi(n
     ,n)
        open(10,file='orbit.d',status='unknown')
        write(*,*) 'a,b,c,d'
        read(*,*) aa,bb,cc,dd
        write(*,*) 'ti,tmax,np (number of points)'
        read(*,*)ti,tmax,np

        t0=ti
        m=-1
        xinctime=dabs(tmax-ti)/np
        a(1,1)=aa
        a(1,2)=bb
        a(2,1)=cc
        a(2,2)=dd
        call vapvep(a,n,rr,ri,vr,vi)


       do 22 k=1,2
          m=-1*m
          do 21 j=1,2
             ti=t0
             x(1)=0.d0
```

```
            x(2)=0.d0
            idir=dsign(1.d0,rr(j))
            x(1)=m*vr(1,j)*1.d-6+x(1)
            x(2)=m*vr(2,j)*1.d-6+x(2)
            write (10,*)ti,(x(ii),ii=1,n)
            do 20 i=1,np
               call flow(ti,n,x,idir,xinctime)
               write (10,*)ti,(x(ii),ii=1,n)
 20         continue
            write(10,90)
            write(*,*)ti,'  final t'
            write(*,*)(x(i),i=1,n),'  final point'
 21      continue
 22   continue

      do 24 j=1,12
         ti=t0
         write(*,*) 'Initial condition x(1),x(2)'
         read(*,*) (xi(i),i=1,n)
c         if (tmax.ge.ti)then
c           idir=1
c         else
          idir=-1
c         endif
         do i=1,n
            x(i)=xi(i)*1.d-6
         enddo
         write(*,*)ti,'   initial t, initial cond:'
         write(*,*)(x(i),i=1,n)
         write (10,*)ti,(x(ii),ii=1,n)
         do 23 i=1,np
            call flow(ti,n,x,idir,xinctime)
            write (10,*)ti,(x(ii),ii=1,n)
 23      continue
         write(10,90)
         write(*,*)ti,'  final t'
         write(*,*)(x(i),i=1,n),'  final point'
 24   continue
 90   format()
      end

      subroutine flow(t,n,x,idir,xinctemps)
      IMPLICIT REAL*8 (A-H,O-Z)
      dimension x(n)
      tmax=t+idir*xinctemps
       hab=0.1e-16
       hre=0.1e-16
       pabs=dlog10(hab)
       prel=dlog10(hre)
```

```
      istep=1
      ht=0.d0
1       CALL taylor_f77_eq_ls_(t,x,idir,istep,pabs,prel,
     & tmax,ht,iordre,ifl)
      if (idir.eq.1.and.t.lt.tmax)go to 1
      if (idir.eq.-1.and.t.gt.tmax)go to 1
      if (dabs(t-tmax).le.1.d-13)return
      write(*,*)'problems in taylor'
      stop
      return
      end
```