F. Méot
BNL C-AD, SBU-SUNY

COMPUTERLAB, EXERCISE 1.2.1-1, SOLUTION

Abstract

Exercise 1.2.1-1 tests the convergence of the numerical solution of the equation of motion, subject to accuracy of the field model (density of the field map mesh, analytical model instead), and subject to the convergence of the numerical integration of the motion ODE (integration step size).

Contents

1 1.2.1-1.a - Change the density of the field map mesh, and change the step size

The fortran program page 3 constructs the required map of a field distribution in cylindrical coordinates, $\mathrm{B_y(r,\theta)}$.

It can be copy-pasted, compiled, and run. Get the executable doing: *gfortran -o geneSectorMap geneSectorMap.f*

The field map it generates is in geneSectorMap.out.

The gnuplot file that generates Fig. 1 can be found in page 4

The gnuplot file that generates Fig. 2 can be found in page 5

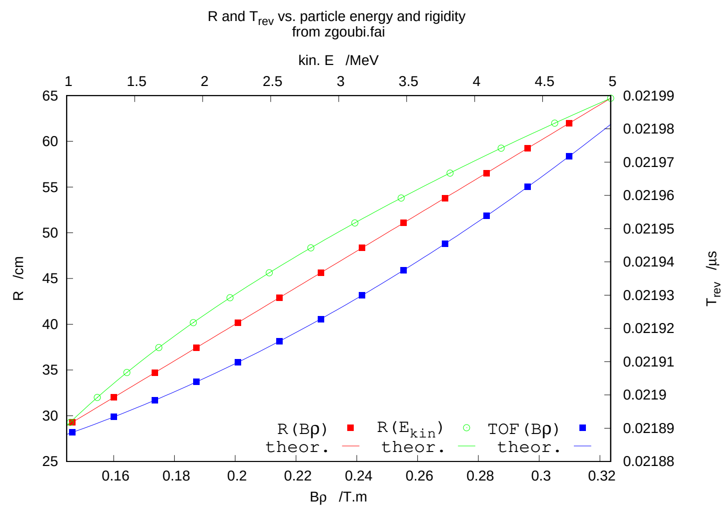

Figure 1: Numerical and theoretical R and $\mathrm{T_{rev}}$.

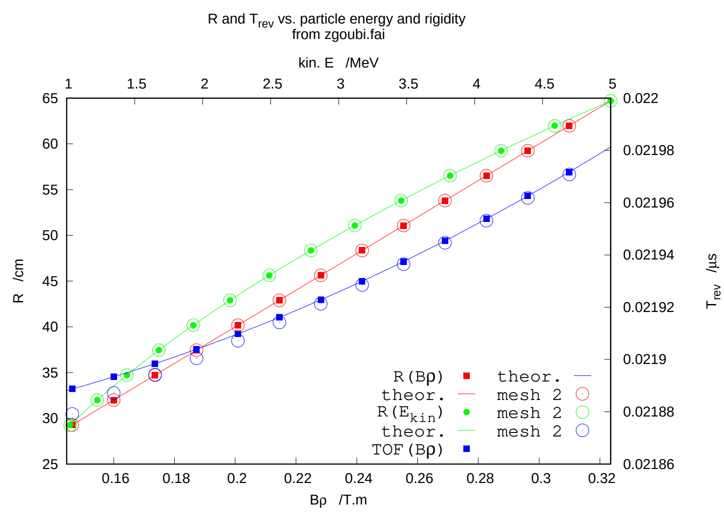

Figure 2: Changing the number of nodes of the mesh from a large value down to a minimum, 3×3, has no effect; this is because the field is constant, here. Instead, clear change in the numerical $\mathrm{T_{rev}}$, due to a large step size $\Delta \mathrm{s} = 10\,\mathrm{cm}$. This results from a change of the circumference of the orbit, thus of R (requires zooming in on R, or plotting the difference between zgoubi.fai and zgoubi_2.fai - modify the gnuplot file accordingly, or use linux command “diff -n”).

A fortran program which generates a 60 deg mid-plane field map

```
      implicit double precision (a-h,o-z)
      parameter (pi = 4.d0*atan(1.d0))

C------------ Hypothesis :
C Total angle extent of the field map. Can be changed, e.g., to 360, 0r 60 deg, or else.
      AT = 60.d0  /180.d0*pi
C Radial extent of the field map
      Rmi = 1.d0   ! cm
      Rma = 76.d0   ! cm
C Take RM=50 cm reference radius, as this (arbitray) value is found in other exercises
      RM = 50.d0
C dR is the radial distance between two nodes, good starting point is dR = 0.5 cm
      dR = 0.5d0            ! cm, mesh step in radius, approximate: allows getting NR
C      dR = (Rma-Rmi)/2.d0      ! field map will have only 3 nodes in R !!
C dX=RM*dA is the arc length between two nodes along R=RM arc, given angle increment dA
C A good starting point (by experience) is dX a few mm, say ~0.5 cm
      dX = 0.5d0             ! cm, mesh step at RM, approximate: allows getting NX
C      dX = AT/2*RM        ! field map will have only 3 nodes in A !!

C------------ Outcomes :
      NR = NINT((Rma - Rmi) / dR) +1
      dR = (Rma - Rmi) / dble(NR -1)   ! make sure (NR-1)*dR == Rma-Rmi
C dX=RM*dA is the arc length between two nodes along R=RM arc, given angle increment dA
      NX = NINT(RM*AT / dX)  +1
      dX = RM*AT / DBLE(NX - 1)  ! exact mesh step at RM, corresponding to NX
      dA = dX / RM               ! corresponding delta_angle
      A1 = 0.d0 ; A2 = AT
C-----------------------------------------------

      BY = 0.d0 ; BX = 0.d0 ; Z = 0.d0
      BZ = 5.d0  ! kG

      open(unit=2,file='geneSectorMap.out')
      write(2,*) Rmi,dR,dA/pi*180.d0,dZ,
     >'     !  Rmi/cm, dR/cm, dA/deg, dZ/cm'
      write(2,*) '# Field map generated using geneSectorMap.f '
      write(2,fmt='(a)') '# AT/rd,  AT/deg, Rmi/cm, Rma/cm, RM/cm,'
     >//' NR, dR/cm, NX, dX/cm, dA/rd : '
      write(2,fmt='(a,1p,5(e16.8,1x),2(i3,1x,e16.8,1x),e16.8)')
     >'# ',AT, AT/pi*180.d0,Rmi, Rma, RM, NR, dR, NX, dX, dA
      write(2,*) '# For TOSCA: ',NX,NR,' 1 22.1 1.  !IZ=1 -> 2D ; '
     >//'MOD=22 -> polar map ; .MOD2=.1 -> one map file'
      write(2,*) '# R*cosA (A:0->360), Z==0, R*sinA, BY, BZ, BX '
      write(2,*) '# cm                 cm    cm      kG  kG  kG '
      write(2,*) '# '

      do jr = 1, NR
        R = Rmi + dble(jr-1)*dR
        do ix = 1, NX
          A = A1 + dble(ix-1)*dA
C          write(2,fmt='(1p,6(e16.8),a)')  R, Z, A, BR, BZ, BA
          X = R * sin(A)
          Y = R * cos(A)
          write(2,fmt='(1p,6(e16.8),2(1x,i0))') Y,Z,X,BY,BZ,BX,ix,jr
        enddo
      enddo

      stop ' Job complete ! Field map stored in geneSectorMap.out.'
      end
```

Plot R and T_{rev}, using gnuplot

```
set tit "R and T_{rev} vs. particle energy and rigidity \n from zgoubi.fai \n"

set xtics nomirror
set x2tics nomirror
set ytics nomirror
set y2tics nomirror

set xlabel 'B{/Symbol r}   /T.m'
set x2label 'kin. E   /MeV'
set ylabel 'R   /cm'
set y2label 'T_{rev}   /{/Symbol m}s'

set key font "roman, 18"
set key maxrow 2
set key samplen 1
set key b r

#set logscale y

### electron :
#am = 0.511
#### proton :
am = 938.27203
m2cm = 100.
MeV2eV = 1e6
s2mus = 1e6
c = 2.99792458e8
B = 0.5   # [T]
pi = 4. * atan(1.)
Nsector = 6.

# Theory:
rho(x) = m2cm * x/B    # [cm]
tof(x) = (2. *pi *x/B/Nsector) / ((x*c)/sqrt((x*c)**2 + (am*MeV2eV)**2)) /c *s2mus   # [cm]
rho2(x) = (sqrt((x+am)**2 - am**2)/B/c) *MeV2eV *m2cm

Ekmi = 1
Ekma = 5
Emi = Ekmi + am
Ema = Ekma + am
Brmi = sqrt(Emi**2-am**2)*MeV2eV/c
Brma = sqrt(Ema**2-am**2)*MeV2eV/c

set xrange [Brmi:Brma]
set x2range [Ekmi:Ekma]

plot  \
   'zgoubi.fai' u (sqrt($25**2-$29**2)*MeV2eV/c):10 w p pt 5 ps 1. lc rgb "red"  tit "R(B{/Symbol r})",\
    rho(x) w l lc rgb "red" tit 'theor.' ,\
   'zgoubi.fai' u ($25-$29):10 axes x2y1 w p pt 6 ps 1 lc rgb "green" tit "R(E_{kin})" ,\
    rho2(x) axes x2y1 w l lc rgb "green" tit 'theor.' ,\
   'zgoubi.fai' u (sqrt($25**2-$29**2)*MeV2eV/c):15 axes x1y2 w p pt 5 ps 1. lc rgb "blue"  tit "TOF(B{/Symbol r})",\
    tof(x) axes x1y2 w l lc rgb "blue" tit 'theor.'

     set terminal postscript eps blacktext color  enh
       set output "gnuplot_zgoubi.fai_RandTOFvsBrho.eps"
       replot
       set terminal X11
       unset output
pause 1
exit
```

Plot R and $\mathrm{T}_{\mathrm{rev}}$, superposing cases with low mesh density, or large integration step size

```
set tit "R and T_{rev} vs. particle energy and rigidity \n from zgoubi.fai \n"

set xtics nomirror
set x2tics nomirror
set ytics nomirror
set y2tics nomirror

set xlabel 'B{/Symbol r}   /T.m'
set x2label 'kin. E   /MeV'
set ylabel 'R   /cm'
set y2label 'T_{rev}   /{/Symbol m}s'

set key font "roman, 18"
set key maxrow 6
set key samplen 1
set key b r

#set logscale y

### electron :
#am = 0.511
#### proton :
am = 938.27203
m2cm = 100.
MeV2eV = 1e6
s2mus = 1e6
c = 2.99792458e8
B = 0.5   # [T]
pi = 4. * atan(1.)
Nsector = 6.

# Theory:
rho(x) = m2cm * x/B    # [cm]
tof(x) = (2. *pi *x/B/Nsector) / ((x*c)/sqrt((x*c)**2 + (am*MeV2eV)**2)) /c *s2mus   # [cm]
rho2(x) = (sqrt((x+am)**2 - am**2)/B/c) *MeV2eV *m2cm

Ekmi = 1
Ekma = 5
Emi = Ekmi + am
Ema = Ekma + am
Brmi = sqrt(Emi**2-am**2)*MeV2eV/c
Brma = sqrt(Ema**2-am**2)*MeV2eV/c

set xrange [Brmi:Brma]
set x2range [Ekmi:Ekma]

plot  \
   'zgoubi.fai' u (sqrt($25**2-$29**2)*MeV2eV/c):10 w p pt 5 ps 1. lc rgb "red"  tit "R(B{/Symbol r})",\
    rho(x) w l lc rgb "red" tit 'theor.' ,\
   'zgoubi.fai' u ($25-$29):10 axes x2y1 w p pt 7 ps 1 lc rgb "green" tit "R(E_{kin})" ,\
    rho2(x) axes x2y1 w l lc rgb "green" tit 'theor.' ,\
   'zgoubi.fai' u (sqrt($25**2-$29**2)*MeV2eV/c):15 axes x1y2 w p pt 5 ps 1. lc rgb "blue"  tit "TOF(B{/Symbol r})",\
    tof(x) axes x1y2 w l lc rgb "blue" tit 'theor.' ,\
   'zgoubi_2.fai' u (sqrt($25**2-$29**2)*MeV2eV/c):10 w p pt 6 ps 2. lc rgb "red"  tit "mesh 2",\
   'zgoubi_2.fai' u ($25-$29):10 axes x2y1 w p pt 6 ps 2 lc rgb "green" tit "mesh 2" ,\
   'zgoubi_2.fai' u (sqrt($25**2-$29**2)*MeV2eV/c):15 axes x1y2 w p pt 6 ps 2. lc rgb "blue"  tit "mesh 2"

     set terminal postscript eps blacktext color  enh
       set output "gnuplot_zgoubi.fai_RandTOFvsBrho_repeat.eps"
       replot
       set terminal X11
       unset output
pause 1
exit
```

2 1.2.1-1.b - Use an analytical model of the cyclotron field

3 1.2.1-1.c - Comments

• The solution does not depend on the field map mesh density, this is clear when going to 3 nodes in R times 3 nodes in angle. The reason is that the filed is uniform (the same everywhere), thus field derivatives a zero and the field interpolation at the location of the particle is exact, see Sec. 1.4.2 in Zgoubi Users Guide.

Note: this will no longer be the case when a radial field index is introduced, to insure vertical focusing, in the next sections of the course.

• Step size is critical. This is because the coefficients of the Taylor series that yield the new postion $\vec{\mathrm{R}}(\mathrm{M}_1)$ and the new velocity $\vec{\mathrm{v}}(\mathrm{M}_1)$ change when the step size changes: see Sec. 1.2, and especially Eq. 1.2.4, in Zgoubi Users Guide, the coefficients of the Taylor series are the derivatives of the velocity vector, which are substantial, especially at small radius R in the cyclotron. Thus, taknig too large a $\Delta\mathrm{S}$ value makes the high order terms significant and the Taylor series truncation is fatal to the accuracy (regardless of possible additional issue of radius of convergence of the series).