F. Méot
BNL C-AD
SBU-SUNY, PHY689

COMPUTERLAB, EXERCISE 1.2.2-1, SOLUTION

Abstract

Adding a field index, for transeverse focusing.

Contents

1 1.2.2-1.a - Uniform field with weak focusing index

Zgoubi input data file is given in page 3. A fortran file to generate a map of the field with index is given in page 4. A gnuplot script to plot a trajectory is given page 5.

Field index: to the first order in x (Sec. 1.2.2 in the course, at y=0 and noting $R = R_0 + x$, $\mathrm{B_y(R_0) = B_0}$, $\mathrm{B_y = B}$),

$$\mathrm{B(x) = B_0 + x \left.\frac{\partial B}{\partial R}\right|_{R_0}} \tag{1}$$

With $\mathrm{k = \frac{R_0}{B_0}\frac{\partial B}{\partial r}}$ this yields

$$\mathrm{B(x) = B_0 + \frac{B_0}{R_0} kx} \tag{2}$$

We take k = -0.03: the field decreases slowly with $\mathrm{R = R_0 + x}$. A low k value reduces the radial extent of the cyclotron, for a given maximum energy. For instance, the 5 MeV orbit is at a radius of 75.754671 cm (B=0.3235 T), whereas if k=0 the R=75.754671 cm orbit is for .6.8463 MeV (B=0.3788 T).

On the contrary, a k value closer to -1 causes a rapid decrease of the field with radius resuling in a smaller energy on the maximum radius. With k=-0.15 for instance, the energy at a radius of 75.754671 cm is 0.50 MeV (B=0.1026 T). A larger —k— has however the advantage of stronger focusing, smaller vertical size of the circulating beam.

Injection orbit

Here, in order to be able to use the same injection conditions as in the earlier exercises, we take

$$\mathrm{R_0 = 12.9248888074\,cm, \text{the 200 keV (injection) radius;}\quad B_0 = B(R_0) = 5\,kG}$$

The field decreases slowly with $\mathrm{R = R_0 + x}$, from 5 kG at $\mathrm{B_0}$.

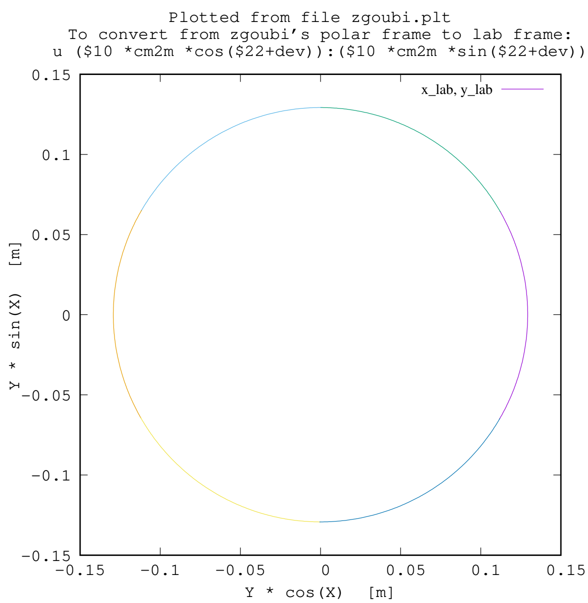

Figure 1: Closed orbit at 200 keV.

Optical sequence in zgoubi, field with radial index

```
Six 60 degree sectors with index.
 'OBJET'
64.62444403717985                                  ! 200keV
2
1 1
12.9248888074 0.  0.   0.   0.  1. 'm'             ! 200keV. R=Brho/B=*/.5
1 1 1 1

 'PARTICUL'   ! This is required only because we want to get the time-of-flight,
PROTON       ! otherwise zgoubi only requires rigidity.

'INCLUDE'
1
60degSector.inc[#S:#E]
 'FAISCEAU'
'INCLUDE'
1
60degSector.inc[#S:#E]
 'FAISCEAU'
'INCLUDE'
1
60degSector.inc[#S:#E]
 'FAISCEAU'

'INCLUDE'
1
60degSector.inc[#S:#E]
 'FAISCEAU'
'INCLUDE'
1
60degSector.inc[#S:#E]
 'FAISCEAU'
'INCLUDE'
1
60degSector.inc[#S:#E]

 'FAISCEAU'

 'FIT'       ! This matching procedure finds the closed orbit radius
1   nofinal
1 30 0 [12.,65.]        ! Variable : Y_0
2  1e-15  9999          ! Penalty; max numb of calls to function
3.1 1 2 #End 0. 1. 0    ! Constraint :  Y_final=Y_0
3.1 1 3 #End 0. 1. 0    ! Constraint :  T_final=T_0

! 'FAISTORE'   ! Particle coordinates at end of match are stored in zgoubi.fai. Ready for plotting!
! zgoubi.fai
! 1
! 'REBELOTE'   ! A do-loop. Repeat the above, after changing particle rigidity to a new value.
! 10 0.2  0 1   ! 20 diffrnt rigidities; I/O options; coordinates as from OBJET; changes follow:
! 1             ! 1 parameter to be changed, in OBJET:
! OBJET 35     1:4.50063899693    ! 0.2 MeV to 1 MeV
 'END'
```

60degSector.inc file

```
60 degree sector with index.
 'OBJET'                                                                                          1
64.62444403717985                                  ! 200keV
2
1 1
12.9248888074 0.  0.   0.   0.  1. 'm'             ! 200keV. R=Brho/B=*/.5
1 1 1 1

 'PARTICUL'   ! This is required only because we want to get the time-of-flight,             2
PROTON       ! otherwise zgoubi only requires rigidity.

 'MARKER' #S                                                                                      3
 'TOSCA'                                                                                          4
0  2
1.  1. 1. 1.
HEADER_8
106  151   1 22.1 1.  !  IZ=1 means 2D map ; MOD=22 means polar coordinates ; .MOD2=.1 means ! one file.
geneSectorMapWithIndex.out
0 0 0 0
2
.1           ! Integration step size.
2
0. 0. 0. 0.

 'MARKER' #E                                                                                      5

 'FAISCEAU'                                                                                       6
 'END'                                                                                            7
```

.

Generate of map of the cyclotron field with index

```
      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 used to build the map, as this (arbitray) value is found in other exercises
      RM = 50.d0
      B0 = 5.d0             ! field at R0
      R0 = 12.9248888074d0  ! cm. 200keV radius at 5kG
      ak = -0.03d0          ! FIEL INDEX
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

      open(unit=2,file='geneSectorMapWithIndex.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
        x = R - R0
        BZ = B0 + B0/R0 * ak * x
        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 geneSectorMapWithIndex.out.'
      end
```

Plot trajectories, using gnuplot

```
set title "Plotted from file zgoubi.plt  \n To convert from zgoubi's \
polar frame to lab frame: \n u ($10 *cm2m *cos($22+dev)):($10 *cm2m *sin($22+dev)) " font "sans, 14"

set key maxcol 1
set key t r

#set logscale y

set xtics mirror font  "sans, 14"
set ytics mirror font  "sans, 14"

set xlabel 'Y * cos(X)  [m]' font  "sans, 14"
set ylabel 'Y * sin(X)  [m]' font  "sans, 14"

cm2m = 0.01
MeV2eV = 1e6
am = 938.27203
c = 2.99792458e8
pi = 4. * atan(1.)

NOEL_1 = 4    #   number of 1st TOSCA in zgoubi,plt (col. 42)
NOEL_2 = 8    #   number of 2nd TOSCA in zgoubi,plt (col. 42)
NOEL_3 = 12   #   number of 3rd TOSCA in zgoubi,plt (col. 42)
NOEL_4 = 16   #   number of 4th TOSCA in zgoubi,plt (col. 42)
NOEL_5 = 20   #   number of 5th TOSCA in zgoubi,plt (col. 42)
NOEL_6 = 24   #   number of 6th TOSCA in zgoubi,plt (col. 42)

dev = 2.*pi/6.

set size ratio -1

plot  \
   'zgoubi.plt' u ($42==NOEL_1 ? $10 *cm2m *cos($22) :1/0):($10 *cm2m *sin($22)) w l tit 'x\_lab, y\_lab' ,\
   'zgoubi.plt' u ($42==NOEL_2 ? $10 *cm2m *cos($22+   dev) :1/0):($10 *cm2m *sin($22+   dev)) w l notit ,\
   'zgoubi.plt' u ($42==NOEL_3 ? $10 *cm2m *cos($22+2.*dev) :1/0):($10 *cm2m *sin($22+2.*dev)) w l notit ,\
   'zgoubi.plt' u ($42==NOEL_4 ? $10 *cm2m *cos($22+3.*dev) :1/0):($10 *cm2m *sin($22+3.*dev)) w l notit ,\
   'zgoubi.plt' u ($42==NOEL_5 ? $10 *cm2m *cos($22+4.*dev) :1/0):($10 *cm2m *sin($22+4.*dev)) w l notit ,\
   'zgoubi.plt' u ($42==NOEL_6 ? $10 *cm2m *cos($22+5.*dev) :1/0):($10 *cm2m *sin($22+5.*dev)) w l notit

     set terminal postscript eps blacktext color  enh  "Times-Sans" 12
       set output "gnuplot_zgoubi.plt_XYLab_addIndex_200keVOrbit.eps"
       replot
       set terminal X11
       unset output

pause 2   # don't change this: needed for proper running of sector180deg
exit
```

2 1.2.2-1.b - Find orbit radius for a series of momenta

The `FAISTORE` and `REBELOTE` commands to be added to the previous exercise, appear commented at the end of the Zgoubi input data list, page 3: just un-comment it. `FAISTORE` stores The data of interest (R and momentum (or rigidity, or energy)) for each `FIT`.

The rigidity on orbit radius R satisfies $\mathrm{B}\rho(\mathrm{R}) = \mathrm{B} \times \mathrm{R}$, which yields

$$\mathrm{B}\rho(\mathrm{R}) = \mathrm{B}_0(1 + \frac{\mathrm{R} - \mathrm{R}_0}{\mathrm{R}_0}\mathrm{k})\,\mathrm{R} \tag{3}$$

A gnuplot script to plot the series of closed orbits (Fig. 2) and the $\mathrm{B}\rho(\mathrm{R})$ dependence (Fig. 3, including Eq. 3) is given page 7.

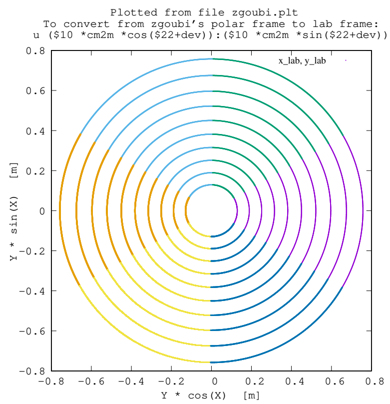

Figure 2: Ten orbits, at rigidities in the range [1:5.0063899693] (relative to 0.06462444403 T m rigidity at 200 keV. The orbits span $12.92 < \mathrm{R} < 75.75467$ cm (still within the field map extent $\mathrm{R}_{\mathrm{max}} = 76$ cm). The plot is from zgoubi.plt data.

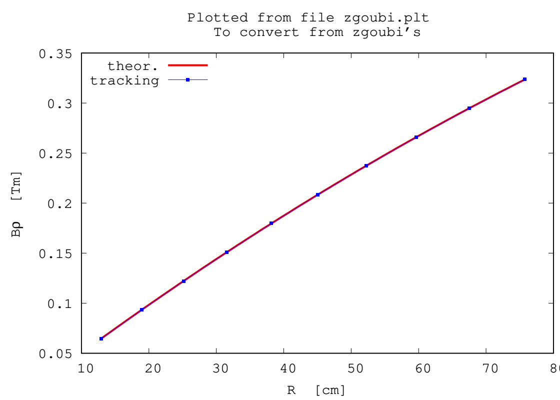

Figure 3: Numerical and theoretical $\mathrm{B}\rho(\mathrm{R})$. The plot is from zgoubi.plt data.

Plot closed orbits at different momenta, using gnuplot

```
set title "Plotted from file zgoubi.plt  \n To convert from zgoubi's \
polar frame to lab frame: \n u ($10 *cm2m *cos($22+dev)):($10 *cm2m *sin($22+dev)) " font "sans, 14"

set key maxcol 1
set key t r

#set logscale y

set xtics mirror font  "sans, 14"
set ytics mirror font  "sans, 14"

set xlabel 'Y * cos(X)  [m]' font  "sans, 14"
set ylabel 'Y * sin(X)  [m]' font  "sans, 14"

cm2m = 0.01
MeV2eV = 1e6
am = 938.27203
c = 2.99792458e8
pi = 4. * atan(1.)

NOEL_1 = 4    #   number of 1st TOSCA in zgoubi,plt (col. 42)
NOEL_2 = 8    #   number of 2nd TOSCA in zgoubi,plt (col. 42)
NOEL_3 = 12   #   number of 3rd TOSCA in zgoubi,plt (col. 42)
NOEL_4 = 16   #   number of 4th TOSCA in zgoubi,plt (col. 42)
NOEL_5 = 20   #   number of 5th TOSCA in zgoubi,plt (col. 42)
NOEL_6 = 24   #   number of 6th TOSCA in zgoubi,plt (col. 42)

dev = 2.*pi/6.

set size ratio -1

plot  \
   'zgoubi.plt' u ($42==NOEL_1 ? $10 *cm2m *cos($22) :1/0):($10 *cm2m *sin($22)) w p ps .2 tit 'x\_lab, y\_lab' ,\
   'zgoubi.plt' u ($42==NOEL_2 ? $10 *cm2m *cos($22+   dev) :1/0):($10 *cm2m *sin($22+   dev)) w p ps .2 notit ,\
   'zgoubi.plt' u ($42==NOEL_3 ? $10 *cm2m *cos($22+2.*dev) :1/0):($10 *cm2m *sin($22+2.*dev)) w p ps .2 notit ,\
   'zgoubi.plt' u ($42==NOEL_4 ? $10 *cm2m *cos($22+3.*dev) :1/0):($10 *cm2m *sin($22+3.*dev)) w p ps .2 notit ,\
   'zgoubi.plt' u ($42==NOEL_5 ? $10 *cm2m *cos($22+4.*dev) :1/0):($10 *cm2m *sin($22+4.*dev)) w p ps .2 notit ,\
   'zgoubi.plt' u ($42==NOEL_6 ? $10 *cm2m *cos($22+5.*dev) :1/0):($10 *cm2m *sin($22+5.*dev)) w p ps .2 notit

     set terminal postscript eps blacktext color  enh  "Times-Sans" 12
       set output "gnuplot_zgoubi.plt_ex1221b_orbits.eps"
       replot
       set terminal X11
       unset output

pause 2   # don't change this: needed for proper running of sector180deg
exit
```

3 1.2.2-1.c - Transverse motion of a particle around the closed orbit

We consider the axial motion for simplicity: the vertical component of the closed orbit is zero, this makes plotting simpler (instead, plotting the radial motion with respect to the horizontal closed orbit requires a 'paste' command in gnuplot, and two zgoubi.plt files to be pasted: the particle zgoubi.plt file, and the closed orbit one).

Eq. 3 can be used (inversed, or by successive approximations) to determine R for a 1 MeV particle. Another possibility is to run a `FIT` with zgoubi, as in the exercise above, for $\mathrm{B}\rho 0.14453544956\,\mathrm{T\,m}$. The latter uses the input file of page 10 which yileds $\mathrm{R}(\mathrm{E} = 1\,\mathrm{MeV}) = 30.107900\,\mathrm{cm}$.

Fig. 4 shows the vertical amplitude of the motion over a few turns. Stronger focusing ($|\mathrm{k}|$ closer to 1) results in smaller motion amplitude, the particle is maintained closer to the reference orbit.

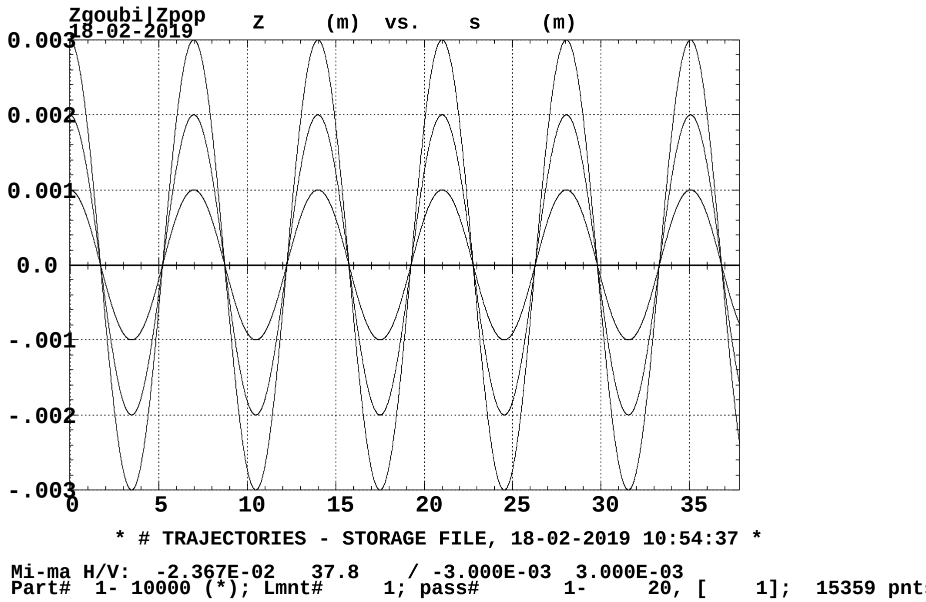

Figure 4: Axial motion around the median plane (the vertical closed orbit lies in that plane). Left: k=-0.03 (1 MeV orbit radius is R=30.1079 cm); right: k=-0.06 (1 MeV orbit radius is R=***** cm). Three particles are tracked, with intial vertical angles, respectively, $\mathrm{P}_0 = 0.1,\ 0.2,\ 0.3\,\mathrm{mrad}$. Anticipating on next lectures: this plot indicates that motion amplitude around the ring is proportional to P_0. Note also that the number of oscillations is less than the number of turns (the vertical wave number is less than 1 and can be infered from this plot).

4 1.2.2-1.d - Magnetic field along particle trajectory

Fig. 5 shows the vertical component of the magnetic field experienced in the course of the vertical motion. Thes plots use zpop.

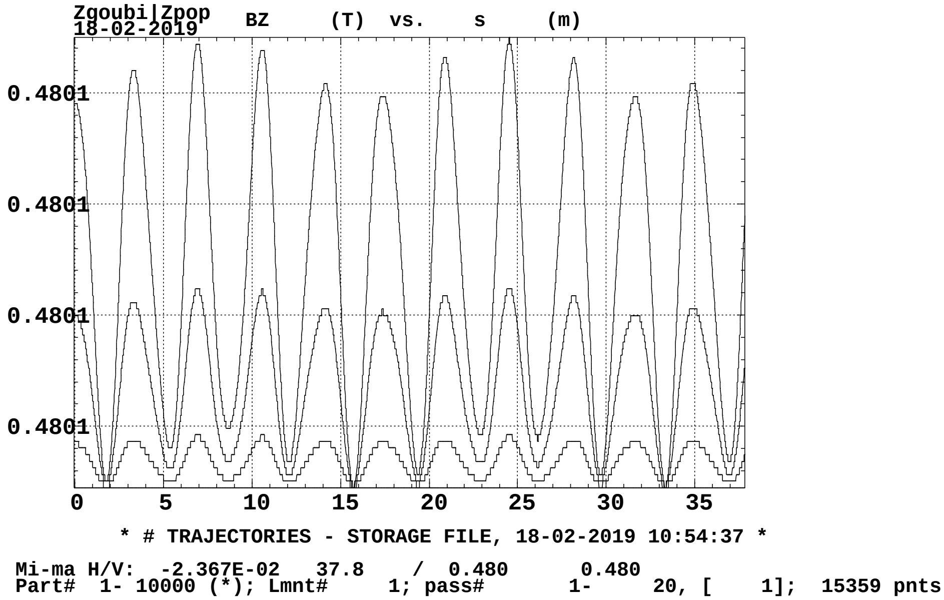

Figure 5: Magnetic field experienced by the 3 particles, in the course of their motion. Explain this and check agaisnt theoretical expectation from Eq. 2.

Input file to find the 1 MeV orbit radius

```
60 degree sector with index. Find Y_0 for or 1 MeV. Brho/Brho_ref = 2.2365445724044024.
 'OBJET'
64.62444403717985                                    ! 200keV reference rigidity.
2
1 1
30.107900 0. 0. 0. 0. 2.2365445724044024 'm'         ! 1 MeV proton -> Brho/Brho_ref = 2.2365445724044024;
1 1 1 1                                              ! Y_0 here resulted from FIT, below.
!! 12.9248888074 0. 0. 0. 0. 2.2365445724044024 'm'  ! Initial (arbitray, closeby) Y_0, before running FIT.
'PARTICUL'   ! This is required only because we want to get the time-of-flight,
PROTON       ! otherwise zgoubi only requires rigidity.

'INCLUDE'
1
60degSector.inc[#S:#E]
 'FAISCEAU'

 'FIT'                  ! This matching procedure finds the closed orbit radius.
1   nofinal
1 30 0 [12.,65.]        ! Variable : Y_0.
2  1e-15  9999          ! Penalty; max numb of calls to function.
3.1 1 2 #End 0. 1. 0    ! Constraint :  Y_final=Y_0.
3.1 1 3 #End 0. 1. 0    ! Constraint :  T_final=T_0.

 'FAISTORE'   ! Particle coordinates at end of match are stored in zgoubi.fai.
zgoubi.fai
1
 'END'
```

Input file to compute particle motion around the 1 MeV closed orbit

```
Six 60 degree sectors with index. Trak paraxial vertical motion at fixed energy.
 'OBJET'
64.62444403717985                                    ! 200keV reference rigidity.
2
4 1
30.107900 0. 0.  0. 0. 2.2365445724044024 'm'         ! 1 MeV closed orbit. Three paraxial rays:
30.107900 0. 0.1 0. 0. 2.2365445724044024 'm'         ! intial vertical angle P_0=0.1 mrad;
30.107900 0. 0.2 0. 0. 2.2365445724044024 'm'         ! intial vertical angle P_0=0.2 mrad;
30.107900 0. 0.3 0. 0. 2.2365445724044024 'm'         ! intial vertical angle P_0=0.3mrad.
1 1 1 1

'PARTICUL'   ! This is required only because we want to get the time-of-flight,
PROTON       ! otherwise zgoubi only requires rigidity.

'INCLUDE'
1
60degSector.inc[#S:#E]
'INCLUDE'
1
60degSector.inc[#S:#E]
'INCLUDE'
1
60degSector.inc[#S:#E]
'INCLUDE'
1
60degSector.inc[#S:#E]
'INCLUDE'
1
60degSector.inc[#S:#E]
'INCLUDE'
1
60degSector.inc[#S:#E]
 'FAISCEAU'

'REBELOTE'
 19  0.1  99                ! A 20-turn tracking.

 'END'
```