TutMo3: FFA Tutorial

David Kelliher & Cédric Hernalsteens

Zgoubi Workshop, Boulder CO
26/08/2019

Tutorial Remit

- This tutorial will cover the “classical” scaling FFAs – namely the radial and spiral sector types.
- Students will learn how to put together an FFA, find the closed orbit and optics, track with acceleration and (if time allows) calculate the dynamic aperture.

Exercise 0 - elements

FFA magnet geometry, r^k field profile, fringe field etc.

FFAG (Radial Sector) – Geometry

The dimensioning of the magnet is defined by

AT	:	total angular aperture
RM	:	mean radius used for the positioning of field boundaries
ACN_i	:	arbitrary inner angle, used for EFBs positioning
ω	:	azimuth of an EFB with respect to ACN
θ	:	angle of an EFB with respect to its azimuth (wedge angle)
R_1, R_2	:	radius of curvature of an EFB
U_1, U_2	:	extent of the linear part of an EFB

N: number of magnets

Locate these parameters in kek_radialFFA.dat with the help of the manual (p234).

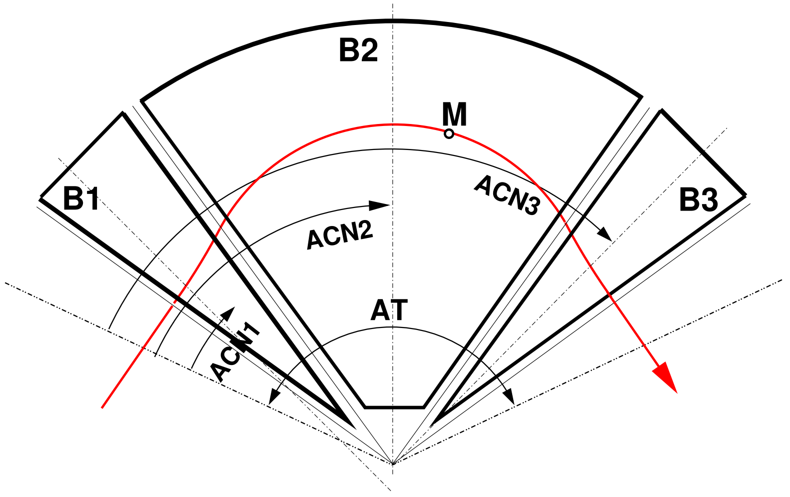


Figure 31: Definition of a dipole N-tuple ($N = 3$, a triplet here) using the *DIPOLES* or *FFAG* procedures.

Descendent of AIMANT like DIPOLE, DIPOLE-S etc.
Polar coordinates – no CHANGREF needed!

FFAG-SPI (Spiral Sector) – Geometry

The dimensioning of the magnet is defined by

AT : total angular aperture
RM : mean radius used for the positioning of field boundaries

ACN_i : arbitrary inner angle, used for EFBs positioning
ω : azimuth of an EFB with respect to ACN
ξ : spiral angle

N: number of magnets

Locate these parameters in raccam_spiralFFA.dat with the help of the manual (p235).

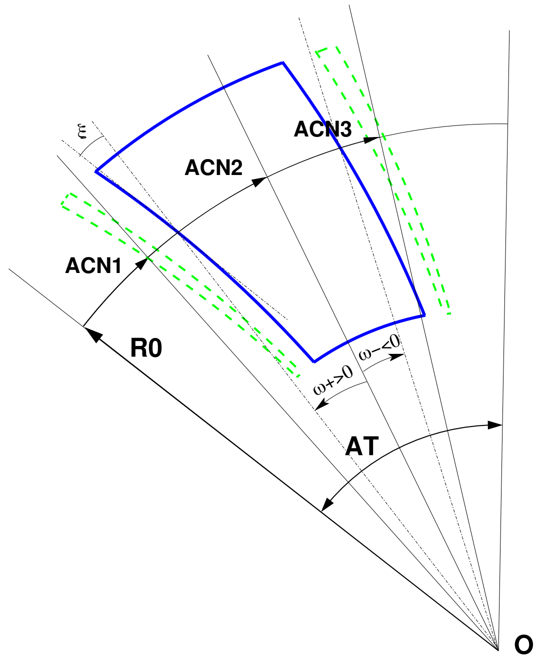
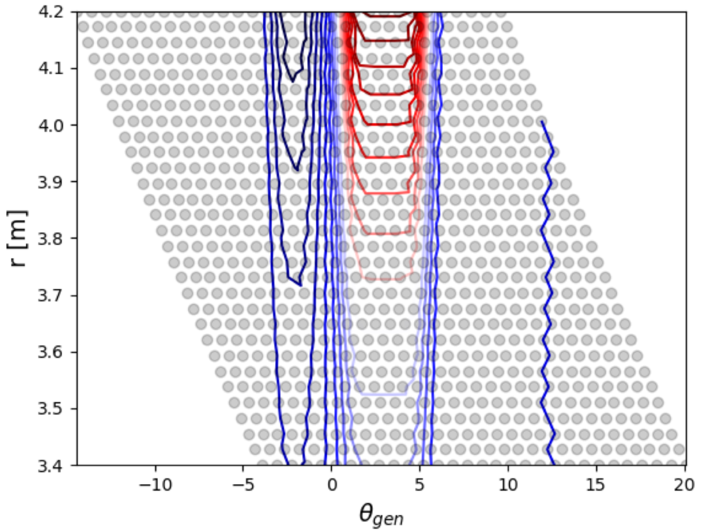


Figure 32: A N-tuple spiral sector FFAG magnet ($N = 3$ here, simulating active field clamps at entrance and exit side of a central dipole).

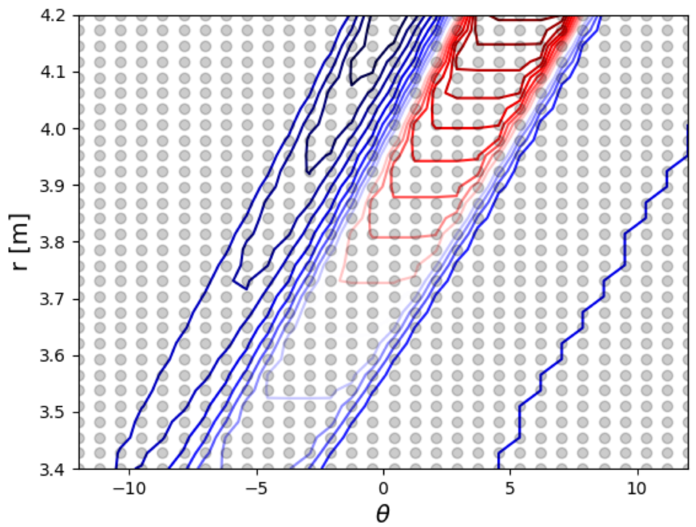
Spiral magnet in various coordinate systems

Generalised polar mesh

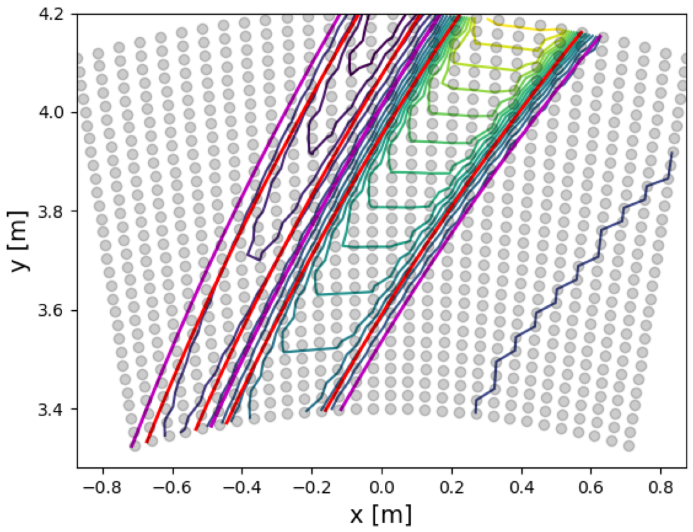
$$\theta_{gen} = \theta - \xi \log(r/r_0)$$



Polar mesh



Cartesian mesh



- Create regular mesh in polar coordinates (r, θ). Transform to (r, θ_{gen}) and (x,y).
- Evaluate midplane field in generalised polar coordinates (r, θ_{gen}) -> $B_z = B_0 (r/r_0)^k F(\theta_{gen})$.

Field profile

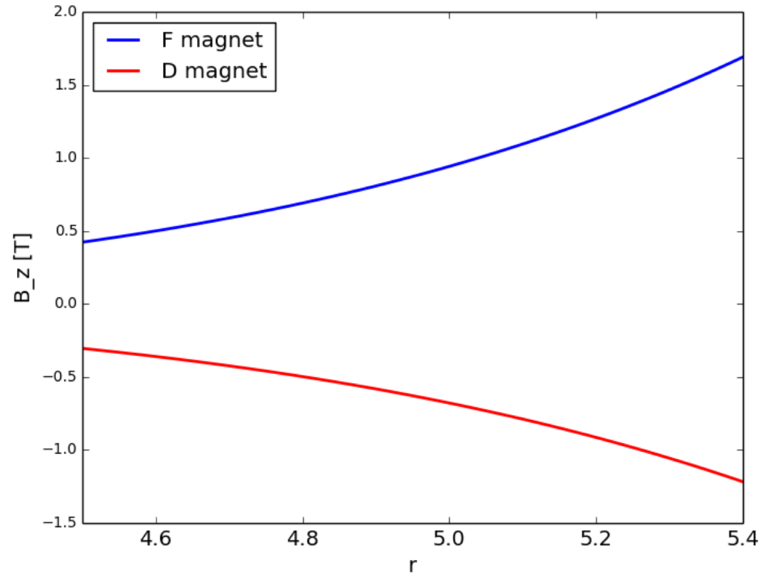
Calculation of the Field From a Single Dipole

The magnetic field is calculated in polar coordinates. At all (R, θ) in the median plane $(z = 0)$, the magnetic field due a single one (index i) of the dipoles of a N-tuple FFAG magnet is written

$$B_{Zi}(R,\theta) = B_{Z0,i}\,\mathcal{F}_i(R,\theta)\;(R/R_M)^{K_i}$$

wherein $B_{Z0,i}$ is a reference field, at reference radius RM_i, whereas $\mathcal{F}(R,\theta)$ is calculated as described below.

Locate the field index, the reference radius and field in a zgoubi input file.



KEK 150 MeV radial FFA

Reference radius, R_M = 5.4m
Field index, K = 7.6
Ref. field, Bz0 (F&D) = 1.69 T, -1.22 T

Fringe field

The fringe field is of Enge type in Zgoubi and is given by

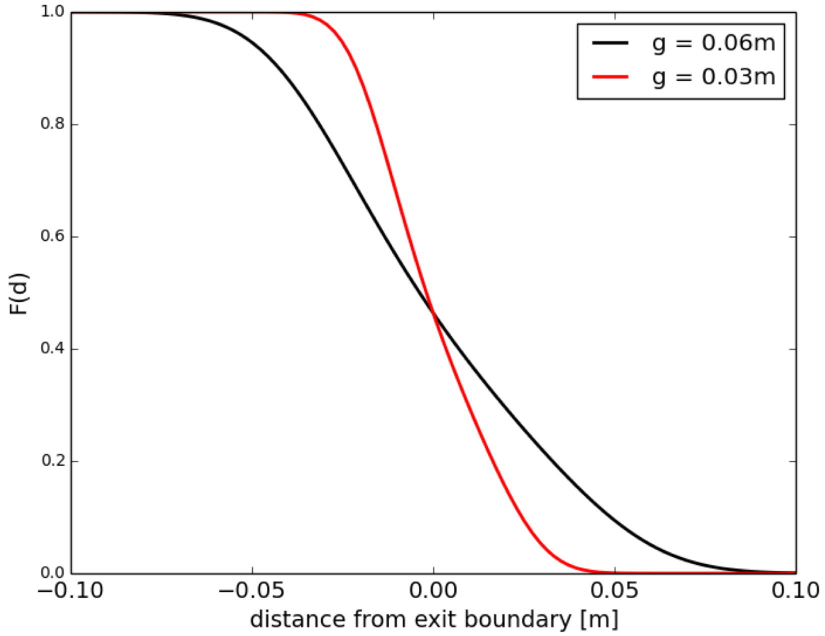
$$F = \frac{1}{1 + \exp P(d)}$$

where d is the distance to the magnet boundary and P(d) is a polynomial

$$P(d) = C_0 + C_1\left(\frac{d}{g}\right) + C_2\left(\frac{d}{g}\right)^2 + C_3\left(\frac{d}{g}\right)^3 + C_4\left(\frac{d}{g}\right)^4 + C_5\left(\frac{d}{g}\right)^5$$

The fringe field extent g varies with radius according to

$$g(R) = g_0\,(RM/R)^{\kappa}$$



Locate g, kappa and the C_i coefficients in a zgoubi input file.

Pyzgoubi – python interface to Zgoubi

- The PyZgoubi interface allows all required elements to be written to a zgoubi.dat file e.g. OBJET, PARTICUL, FAISCNL.

```
ffag_s = FFAGSPI('ffag_spi', N = nffag, AT=angle, RM=rm*cm_,
KIRD=kird, RESOL=resol, XPAS=xpas, KPOS=2, TE=T_mag, TS = T_mag, IL=2)
ffag_s.add(ACN = acn1, BZ_0 = bz_f*kgauss_, K = kindex,
              G0_E = g0*cm_, KAPPA_E = kappa,
              NCE = nc, CE_0 = c0, CE_1 = c1, CE_2 = c2, CE_3 = c3,
              OMEGA_E = omega_f, XI_E = xi,
              G0_S= g0*cm_, KAPPA_S=kappa,
              NCS = nc, CS_0 = c0, CS_1 = c1, CS_2 = c2, CS_3 = c3,
              OMEGA_S = -omega_f, XI_S = xi,
              KAPPA_L = -1)
cell.add(ffag_s)
cell.add(FAISCNL(FNAME='zgoubi.fai'))
```

- PyZgoubi will then run Zgoubi, read the output files and can perform tasks such as finding the closed orbit orbit.

Exercise 1 – closed orbit

Find the closed orbit across the momentum range. A few methods are described.

Closed orbit

- The closed orbit, in a periodic lattice, is the set of transverse coordinates that maps make to itself.
- Method 1 – track a single particle. Take the average x,x' coordinates as the starting point for the next iteration. Repeat until the solution converges.
- Method 2 – use FIT to automate the procedure.
- Method 3 – use PyZgoubi to automate closed orbit finding.

Find the closed orbit across the momentum range in RACCAM.

Exercise 2 – optics

Find the optics at the end of the cell and then throughout the cell.

Optics calculation

- Switch to OBJET mode 5. Transfer matrix generated by ‘MATRIX’ keyword – optics parameters calculated at the end of the cell.

Offset → 1E-4 1E-3 1E-4 1E-3 1E-3 1E-3

Closed orbit → 517.498257090006 -1.7196220340300696e-07 0.0 0.0 0.0 1.0 ‘ ‘

```
'OBJET'
1.839089852668e+03
5
1E-4 1E-3 1E-4 1E-3 1E-3 1E-3
517.498257090006 -1.7196220340300696e-07 0.0 0.0 0.0 1.0 ‘ ‘
…
'FAISCNL'
zgoubi.fai
'MATRIX'
1 11
'END'
```

1 11 ← IFOC=11 (periodic condition)

Optics calculation

- Multiple momenta case

5.N where N is momentum no →

Multiple closed orbits

```
'OBJET'
1.839089852668e+03
5.8
1E-4 1E-3 1E-4 1E-3 1E-3 1E-3
442.9875929615617 2.386261913302348e-08 0 0 0 0.2613492370235746 ' '
460.707511196757 -5.013867901860651e-08 0 0 0 0.3668707745916353 6 ' '
474.3932567996314 -8.379416878329184e-08 0 0 0 0.4723923121596961 ' '
485.60769963153444 -8.81986930895617e-08 0 0 0 0.5779138497277568 ' '
495.1428625098331 -1.1039146190291185e-07 0 0 0 0.6834353872958178 ' '
503.4580211549308 -1.3856365777610592e-07 0 0 0 0.7889569248638785 ' '
510.84428208185363 -1.5649646154703395e-07 0 0 0 0.8944784624319392 ' '
517.498257090006 -1.7196220340300696e-07 0.0 0.0 0.0 1.0 ‘ ‘
...
```

Optics calculation

- Using the optics at the end of the cell and the transfer matrices, find optics parameters throughout cell using the Pyzgoubi function get_twiss_profiles().

Calculate the optics in the RACCAM case.

Exercise 3 – acceleration

Deal with the frequency sweep in a scaling FFA.

RF frequency law

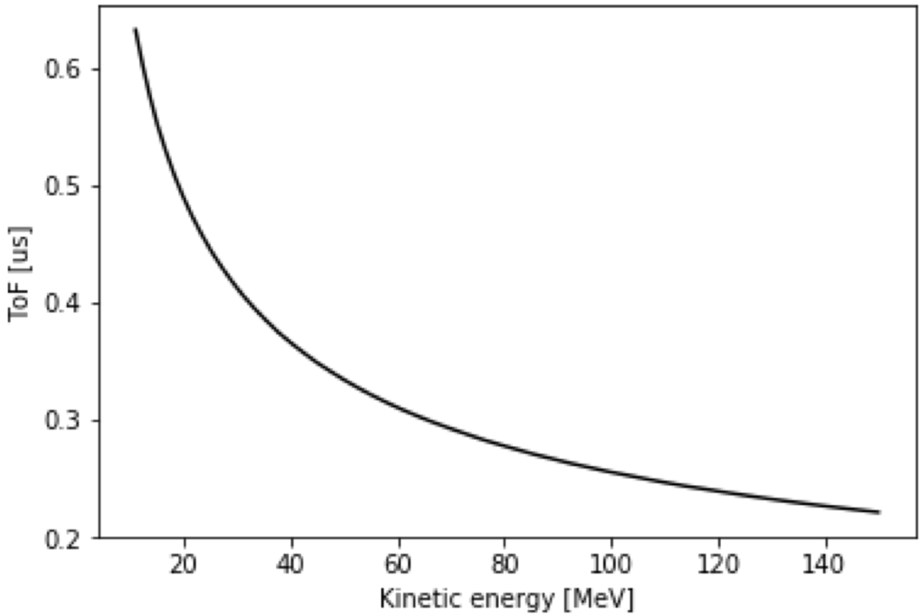
BNL-98791-2012-CP

In order to achieve the acceleration cycle from 12 to 150 MeV, the RF frequency is increased from 1.641 to 4.513 MHz. The synchronous phase $\phi_s = 20°$ and the peak voltage $\hat{V} = 10$ kV are maintained constant, thus about 10000 turns (3.10 ms) are needed. A single cavity is used in the simulations, located in the middle of one of the 12 drifts.

The RF phase $\phi(t) = 2\pi \int_0^t f_{RF}(t)dt$ upon arrival of a particle at the RF gap at time t, is computed by interpolating f_{RF} from the frequency revolution law. The latter might be derived from the scaling law, above, using the revolution time $\tau = 2\pi r/\beta c$ on an orbit with radius r at energy $E = pc/\beta$, namely

$$\tau = \tau_0 \left(\frac{p}{p_0}\right)^{\frac{-k}{k+1}} \frac{E+m}{(E_0+m)}$$

with $E + m$ the particle energy. However, it is as straightforward to compute τ numerically by prior ray-tracing of a few tens of closed orbits taken between 17 and 180 MeV. This in addition guarantees agreement with the zgoubi model (although, as seen in the table below, the formula above on the one hand and the tracking on the other hand happen to lead to extremely small difference in revolution time).



Procedure

- 1. Run *kek_150MeVradialFFA_co2.py* to calculate the TOF across the momentum range and, hence, generate searchCO.outCOs. Copy searchCO.outCOs into the freqlaw_tool subdirectory.
- 2. Go into the freq_tool directory. If an executable isn't present, first compile Et2nf.f. Modify Et2nf.In with the desired RF voltage, synchronous phase and energy range.
- 3. Run the executable. There should now be a zgoubi.freqLaw.In file linked to freqlaw_tool/Et2nf.Out.
- 3. Set the desired number of turns in **kek_phaseEnerg.dat** (in REBELOTE). Run the input file.
- 4. View the output (zgoubi.fai) using the plotFai_accel jupyter script.

Input file parameters

```
'SCALING'    #START
 1   1
CAVITE
  -2
     1   unused scl
     1   unused tim
```

...LATTICE...

```
'CAVITE'
 6   .1
0.      0.      at start : synchr. orbl (m) & Ekin W_s0 (MeV)
 40000.   .349066      Vp (V),  phis (rad)
```

Accelerate in the RACCAM case.