ADVANDED LATTICE MODELLING

François Méot

Brookhaven National Laboratory - Collider-Accelerator Department

Upton, NY, USA

Contents

Foreword

• **By "Advanced", it is essentially meant "based on stepwise numerical integration methods", with some corollaries - some being of major interest !**

- It gives a refined view of charged particle motion through magnetic and electrostatic devices, this helps understanding the physics.

This also allows access to refined optical settings, as it allows to manipulate field models.

- A corollary is that a detailed description of the fields in which particles are moving is required. This is a specificity of the method.

- Specificities of the stepwise ray-tracing method : some of these make it unbeatable, in a number of utilizations, compared to more classical methods of beam optics, for instance spectrometer design, highly non-linear optics, large 6-D particle excursions (kinematic terms...), etc. That's what this lecture is also about

• **There is a couple of areas where stepwise ray-tracing may be less performing, compared to standard techniques (matrix transport, 1-turn map, or other kick-drift methods). Essentially there :**

- CPU time, a potential problem for "long-term tracking" studies

- "symplecticity". A property to keep an eye when using stepwise methods again for "long-term tracking"

• **It can be too sophisticated ! "squashing a fly with a bulldozer". For instance in studying *paraxial* beam line or ring optics.**

1 Introduction

• In this lecture I will

- introduce basic theoretical material in relation with numerical solving of charged particle optics problems,

- with the objective of being able to use them later today, in the software workshop, for the simulation of linear and scaling FFAG cells.

• The lecture is organized in the following way :

1/ I will give a short introduction to stepwise ODE solving methods

3/ We will see how we can simulate, using stepwise ray-tracing methods,

(i) a linear FFAG cell, based on a quadrupole doublet,
(ii) a scaling, KURRI-style dipole triplet cell.

3/ I will introduce to some basic ways, based on ray-tracing, to get

- orbits, global optical parameters : tunes and spectra, chromaticities,
- optical functions inside optical elements,
- multiturn-tracking in a ring made up of these cells,
- dynamical acceptance, etc.

• **The result of this rapid "tour" will be that we will have seen some theoretical material in relation with the understanding of**

(i) what stepwise-based and tracking-based methods bring and some specificities in their manipulation,
(ii) what one can expect, using them,
(iii) basic ingredients entering in the simulation of linear- and scaling-style FFAG cells and rings.

• **A couple more words, about the software workshop later today :**

- We will play with these two types of cells (using the stepwise ray-tracing code Zgoubi) :
  - linear ffag cell,
  - scaling dipole triplet.

- We will work at producing orbits, 1-turn maps, optical functions, motion spectra, DA, as time allows.

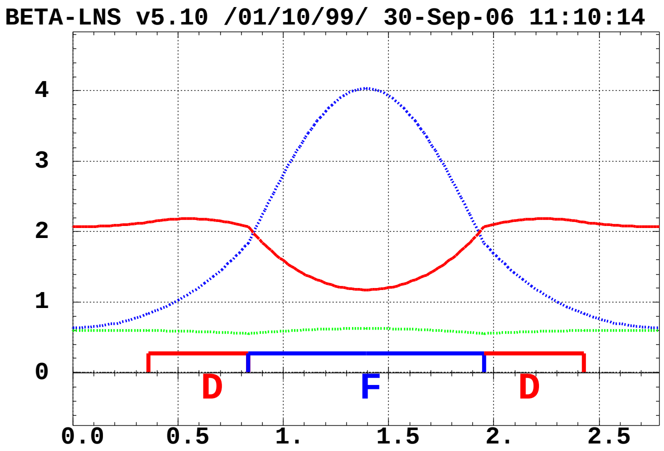
Data analisis from Zgoubi ray-tracing will use the zpop interface, or pyZgoubi (S. Tygier)

Graphics will use zpop, or various simple gnuplot files, or again pyZgoubi

- To conclude with this introduction, WHAT THIS LECTURE IS NOT ABOUT :
  IT IS NOT ABOUT MATRIX (approximate) SOLUTION OF $\frac{dm\vec{v}}{dt} = q\vec{v} \times \vec{B}$.
  WE WON'T DO THAT !

Linear optics : Lorentz equation is simplified to Hill equation

This allows preliminary design steps based on regular matrix methods



First : find a closed orbit ← from the FFAG parameters.

Then : linear approximation about that closed orbit

$$x'' + \frac{1-n}{\rho^2}x = 0, \quad z'' + \frac{n}{\rho^2}z = 0$$

with $\quad n(s) = -\frac{\rho(s)}{B(s)}\frac{dB}{dx} \approx -\frac{\rho}{B}\frac{dB}{dr}$ (scalloping is neglected)

Index $n(s)$ and K in $B = B_0(\frac{r}{r_0})^K$ relate as follows:

$\frac{dB}{dr} = K\frac{B_0}{r_0}(\frac{r}{r_0})^{K-1} = K\frac{B}{r}$ so that $\boxed{K/r = -n/\rho}$

The matrix representing a sector has the form $M = \begin{bmatrix} \cos(s\sqrt{k}) & \frac{1}{\sqrt{k}}\sin(s\sqrt{k}) \\ -\sqrt{k}\sin(s\sqrt{k}) & \cos(s\sqrt{k}) \end{bmatrix}$

with $k = (1-n)/\rho^2$ (radial motion) or $k = n/\rho^2$ (vertical motion)

The geometry provides the wedge angles, hence wedge matrices, M_{Fe1}, M_{Fe2}, M_{De1}, M_{2e2}

The product matrix representing a D-F sector yields the phase advance :

$\cos(\mu) = \frac{1}{2}Tr(M_{Fe2} \times M_F \times M_{Fe1} \times M_{De2} \times M_D \times M_{De1})/2$,...

2 An ODE ?? Where ??

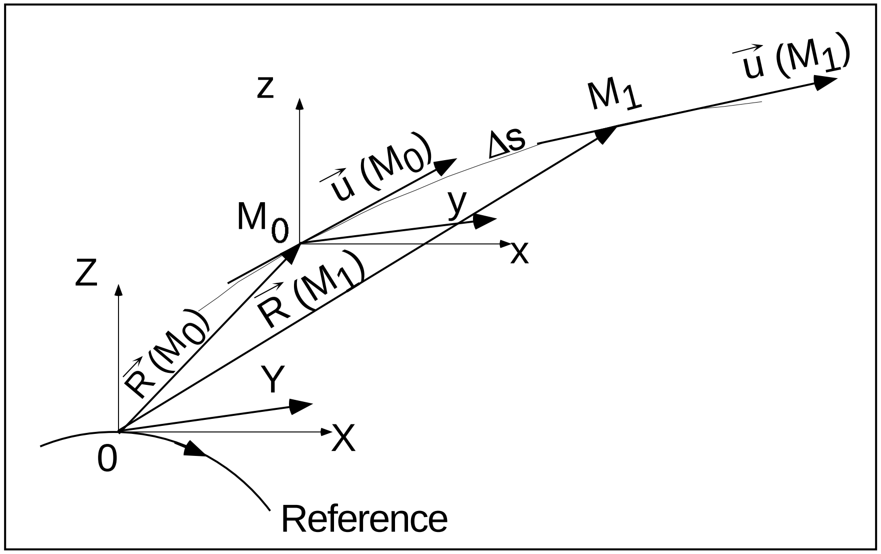
• **Here ! We need to solve an Ordinary Differential Equation, namely, as we know :**

$$\frac{dm\vec{v}}{dt} = q\vec{v} \times \vec{B}$$

For that we want to find a numerical approximation to the solution, using some efficient (accurate and fast, as much as doable) numerical method.

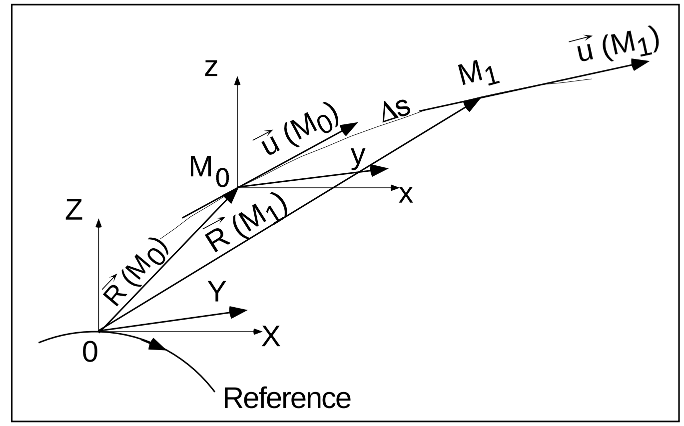
• **Many numerical methods are available, to solve ODEs / push particles :**

Euler,
Taylor series,
Runge-Kutta (very popular),
Multistep ...



Pushing a particle from M_0 to M_1.

• **We will comment on a Taylor series method, as it is the method installed in the ray-tracing code we want to use later today**



Pushing a particle from M_0 to M_1.

- **Zgoubi numerical integrator :**

$$\text{Position} \ : \ \vec{R}(M_1) \approx \vec{R}(M_0) + \vec{u}(M_0)\,\Delta s + \vec{u}'(M_0)\,\frac{\Delta s^2}{2!} + ... + \vec{u}'''''(M_0)\,\frac{\Delta s^6}{6!} \qquad (1)$$

$$\text{Velocity} \ : \ \vec{u}(M_1) \approx \vec{u}(M_0) + \vec{u}'(M_0)\,\Delta s + \vec{u}''(M_0)\,\frac{\Delta s^2}{2!} + ... + \vec{u}'''''(M_0)\,\frac{\Delta s^5}{5!} \qquad (2)$$

(Definitely, no matrix transport there !!)

And as well, when integrating in electrostatic fields :

$$\text{Rigidity} \ : \ (B\rho)(M_1) \approx (B\rho)(M_0) + (B\rho)'(M_0)\Delta s + ... + (B\rho)'''''(M_0)\frac{\Delta s^5}{5!} \qquad (3)$$

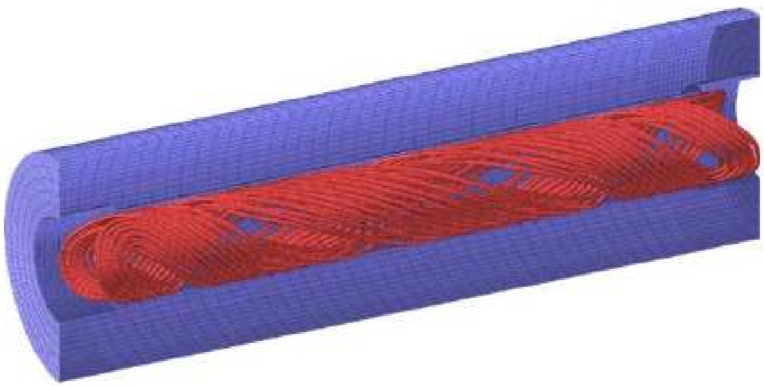
$$\text{Time} \ : \ T(M_1) \approx T(M_0) + T'(M_0)\,\Delta s + T''(M_0)\,\frac{\Delta s^2}{2} + \ ... \ + T'''''(M_0)\,\frac{\Delta s^5}{5!} \qquad (4)$$

Let's see a little more in detail - jump to Zgoubi Users' Guide, page 19

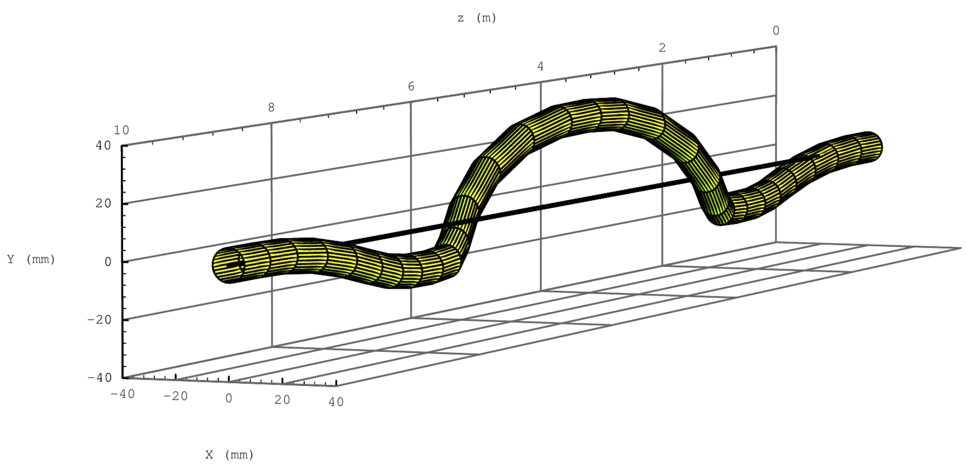
3 Let’s Illustrate the Sophistication With an Example That Has Nothing to do With FFAGs

Taken from real life : Optimization of spin transport through an helical siberian snake

3.1 Helical siberian snake



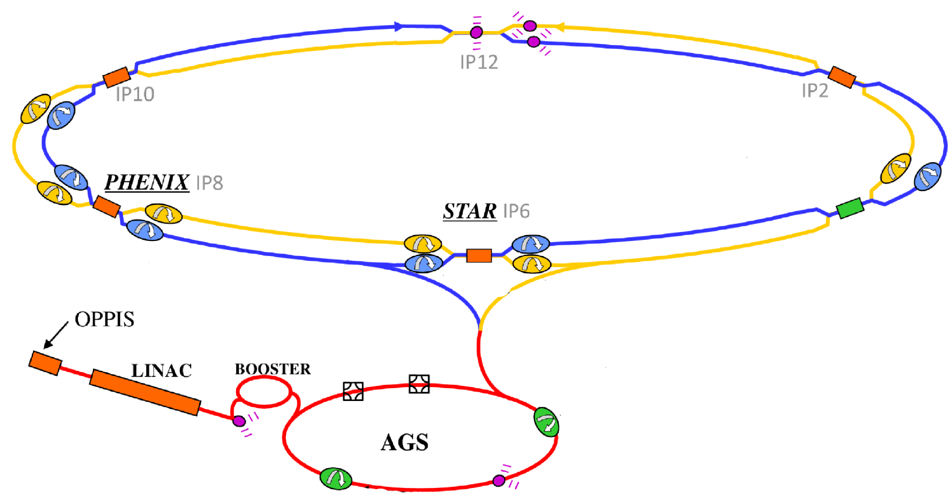
OPERA model of the AGS cold snake : twisted dipole coil + superimposed solenoid coil (not shown) for coupling compensation



Helical trajectory in a (“full”, 180 deg spin rotation) helical dipole

Equations of the helical wiggler field :

$$
\begin{aligned}
B_r &= 2B_0\left[I_0(kr) - (i/kr)I_1(kr)\right]\sin(\theta - kz) \\
B_\theta &= (2B_0/kr)I_1(kr)\cos(\theta - kz) \\
B_z &= -2B_0 J_1(kr)\cos(\theta - kz)
\end{aligned}
\tag{5}
$$



We use such snakes in the AGS (2, “partial snakes”) and in RHIC (2 per ring, “full snakes”), for 250 GeV polarized proton collisions.

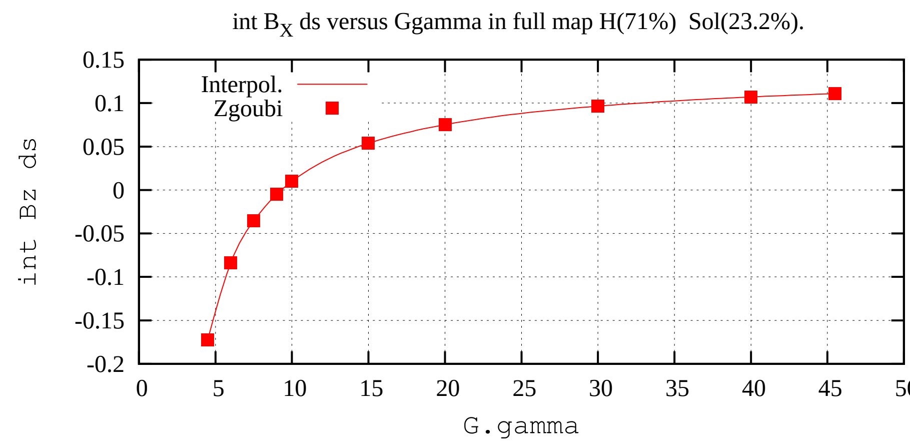
Let's Illustrate With an Example (cont'd)

- **It is necessary to minimize coupling from helical dipole at $G\gamma \approx 9$ (proton E≈4.7 GeV).**

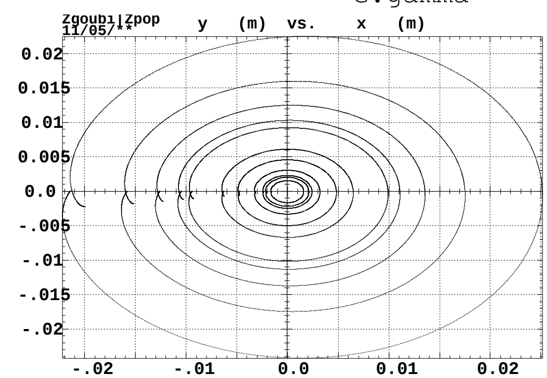
- **This is in order to minimize depolarization effect when proton beam is accelerated through that energy region (depolarizing condition : $G\gamma \pm Q_y =$integer).**

- **Minimizing coupling induced by snake is based on optimal field compensation solenoid.**

Integral $\int B_z(s)ds$ of longitudinal field component along orbit.



Solenoid field set such that $\int B_z(s)ds$ vanishes at $G\gamma \approx 9$.



- **In the mean time, helical trajectory has to remained centered on snake axis. And spin has to z-rotate by defined value, 22 degree.**

- **This requires constraints on helical trajectory across the snake.**

• What a 'FIT' procedure can do (it can do a lot !) :

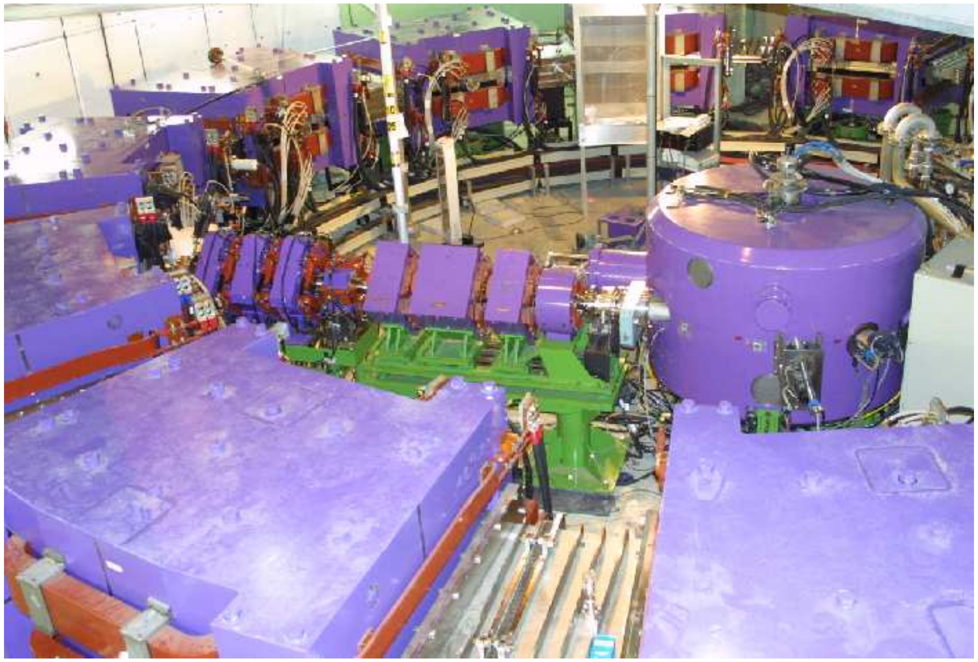
```
 'OBJET'
15.3962411375d3   ! G.gamma=9
2
1     1
-1.031655822  0. 0. 0. 0.    1. 'o'
1 1

 'DRIFT' DRIF   CSNK
-10.
 'TOSCA'            ! Helix + solenoid are defined using 2 OPERA 3-D field maps
0  2
10.15 1. 1. 1.      ! 10.15 yields precession equal to that of full hlx+sol map
HEADER_9            ! 0.960563380 = 68.2% hlx /71
321 29 29  15.2  0.960563380  0.2172656783    !    Cancel sum_BX.ds at Ggamma=9
ags-full-coilv5-x06-rerun2-x071-integral-x5y5z10mm.table
ags-full-sold3-only-nodal-x5y5z10mm-wasactually-integral.table
0 0 0 0
2
.1
2  0.  .0  0.  0.
 'DRIFT' DRIF   CSNK
-10.

 'FIT'
2               ! Two variables
1 30 0  [-1.5,1.5]   ! Vary Y_0 (prtcl 1) (to ensure centering of orbit on snake axis)
6 25 0  .05          ! Vary field of solenoid map (to ensure \int B_z(s) ds ==0)
2   1e-10       ! Two constraints
4.3 1 2 6 0. 1. 0     ! trajectory across snake should be centered, |max_Y| = |min_Y|
4.9 1 1 6 0. .1 0     ! particl should experiences sum(BX.ds)=0 in snake
 'END'
```

4 Back to FFAGs... Simulating FFAG accelerators

4.1 Scaling FFAG

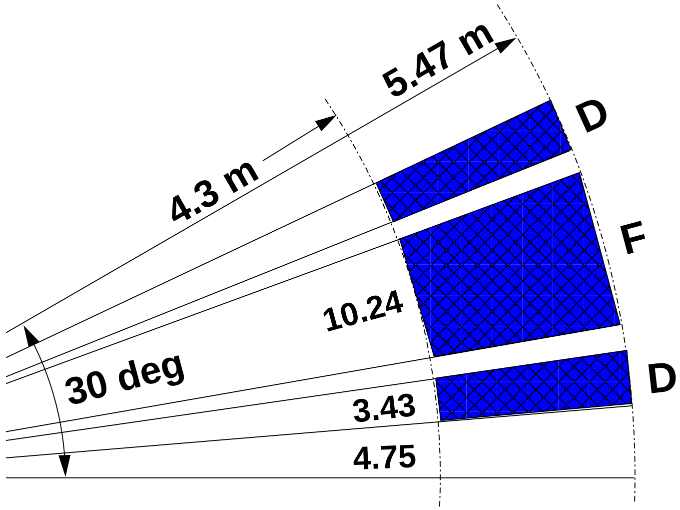


Main parameters :

$E_{inj} - E_{max}$	**MeV**	**12 - 150**
orbit radius	**m**	**4.47 - 5.20**
lattice / K		**DFD $\times$ 12 / 7.6**
β_r / β_z **max.**	**m**	**2.5 / 4.5**
ν_r / ν_z		**3.7 / 1.3**
B_D / B_F	**T**	**0.2-0.78 / 0.5-1.63**
gap	**cm**	**23.2 - 4.2**
RF swing	**MHz**	**1.5 - 4.5**
voltage p-to-p	**kV**	**2**

PoP scaling triplet

4.1.1 Simulation of a scaling dipole triplet

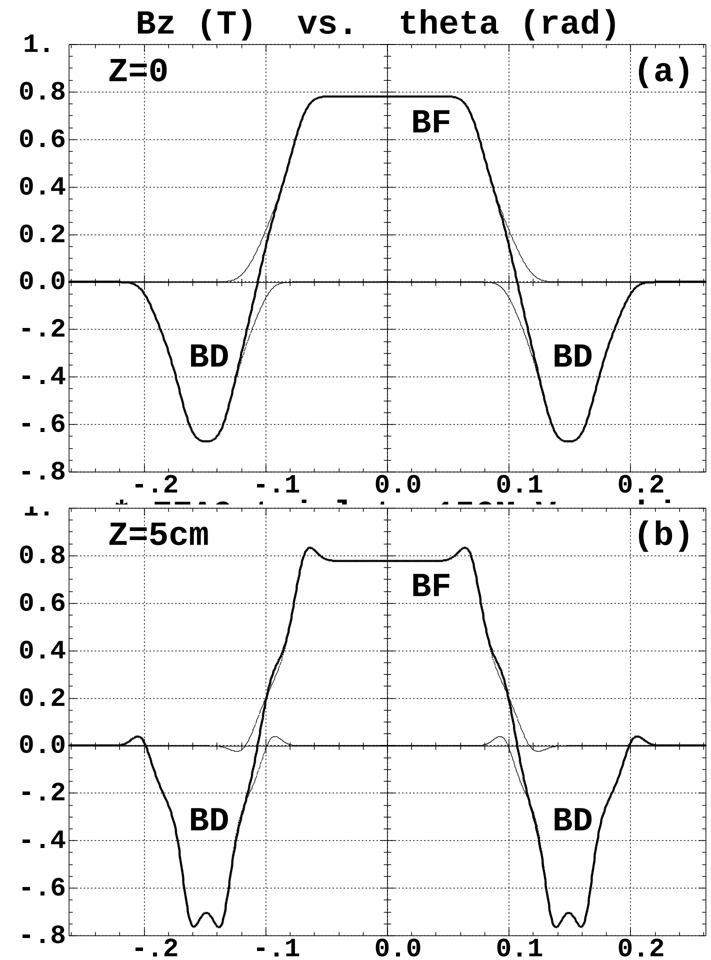


DFD triplet.

• A geometrical model based on the superposition of independent contributions of the N dipoles, to provide the mid-plane field, at all $(r, \theta, z=0)$:

$$B_z(r,\theta) = \sum_{i=1,N} B_{z0,i}\, \mathcal{F}_i(r,\theta)\, \mathcal{R}_i(r)$$

• Field at (r, θ, z), off mid-plane, is obtained by Taylor expansion accounting for Maxwell's equations.



Yes ! from ray-tracing :

field experienced along $r_0 = 4.87$ m arc,

at $\begin{cases} z = 0 \\ z = 5cm \end{cases}$,

in KURRI-style DFD dipole triplet.

4.1.2 How to simulate that dipole N-tuple in a ray-tracing code ?

25 (FFAG keyword) and Fig. 11 p. 88 (define dipole geometry), page 112 (a

(The list of available optical elements can be found in pp. 7, 8 (PART A of the guide, tells how it works) or as well in pp. 177, 178 (PART B of the guide, tells inpput data formatting in zgoubi.dat))

4.1.3 What can be expected from ray-tracing

- **Accurate trajectory computation in arbitrary fields,**

- **And then, from trajectories, everything can be derived :**

  - optical functions

  - and higher order : optical aberrations

  - "multi-turn tracking"

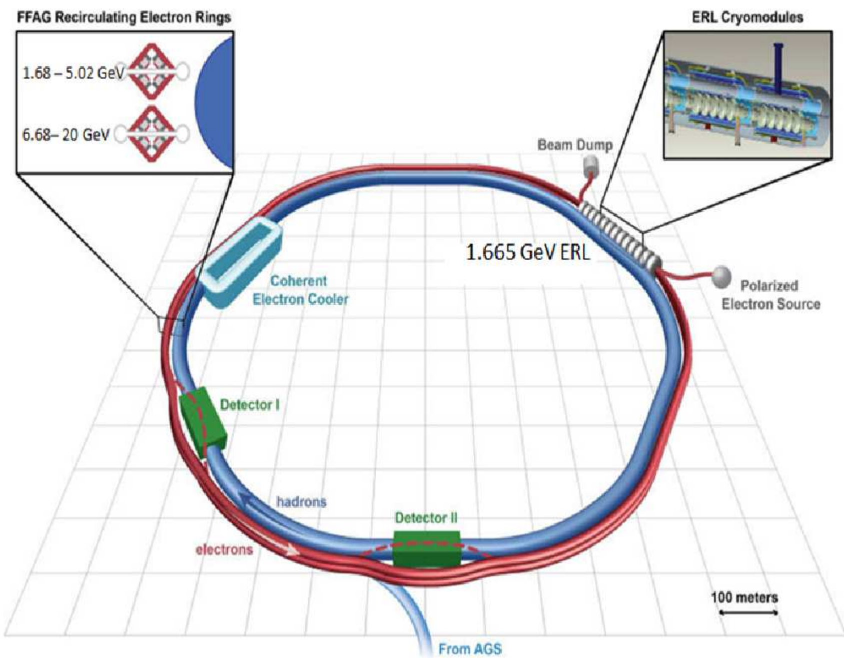
  - phase-space motion

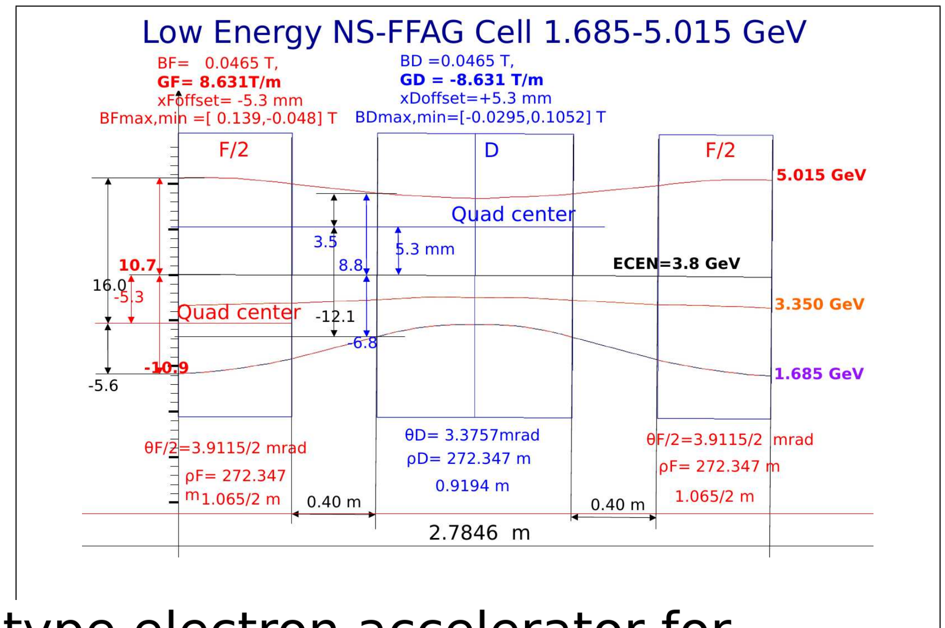
  - motion spectra ("wavenumbers" aka "tunes")

  - etc, etc,

4.1.4 Simulation of a linear FFAG cell

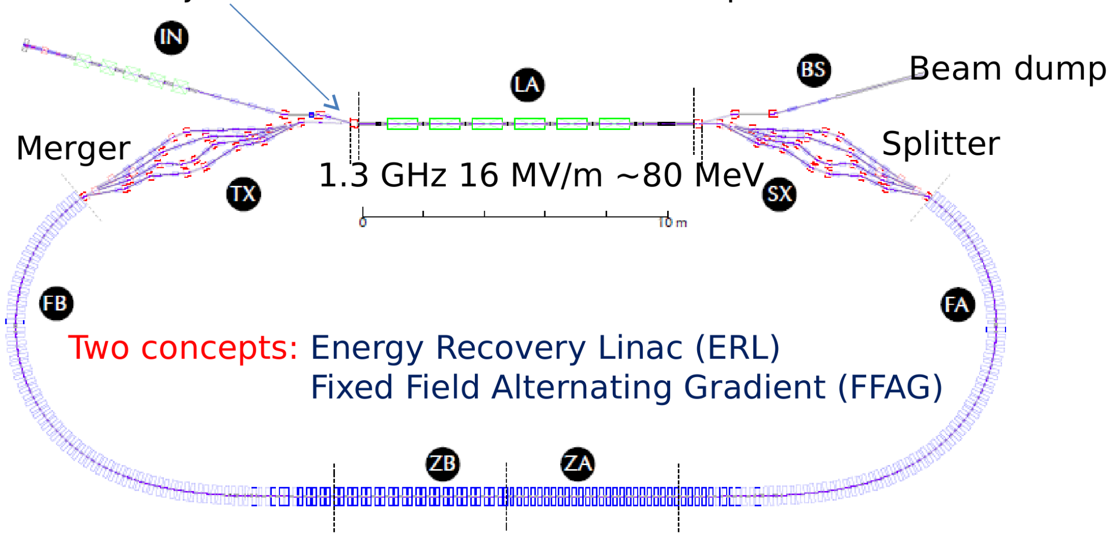
4.2 Linear FFAG





Cβ : a prototype electron accelerator for the eRHIC project

Injection: 5 MeV e-   100 mA   rep-rate 325 MHz

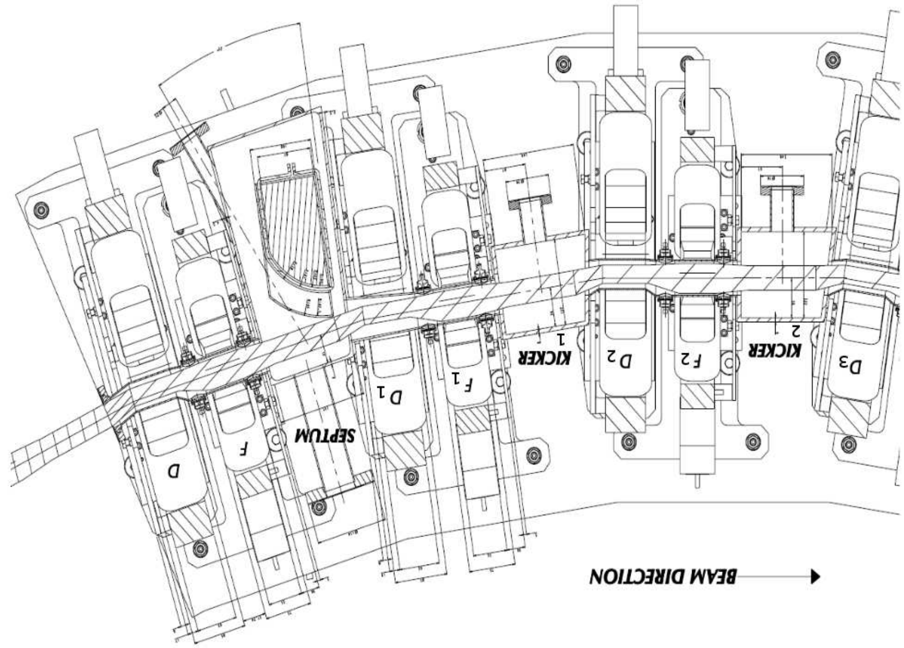


Two concepts: Energy Recovery Linac (ERL)
Fixed Field Alternating Gradient (FFAG)

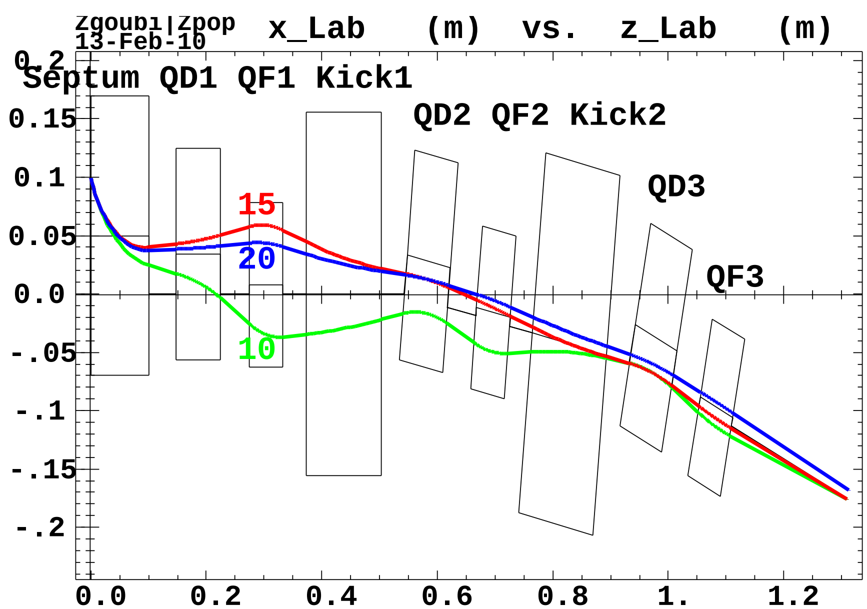
EMMA

4.2.1 EMMA Simulations

EMMA injection section ...



... and its ray-tracing simulation : optical references at three different energies



4.2.2 How to simulate a quadrupole in a ray-tracing code ?

- Straightforward optical element : 'QUADRUPO'

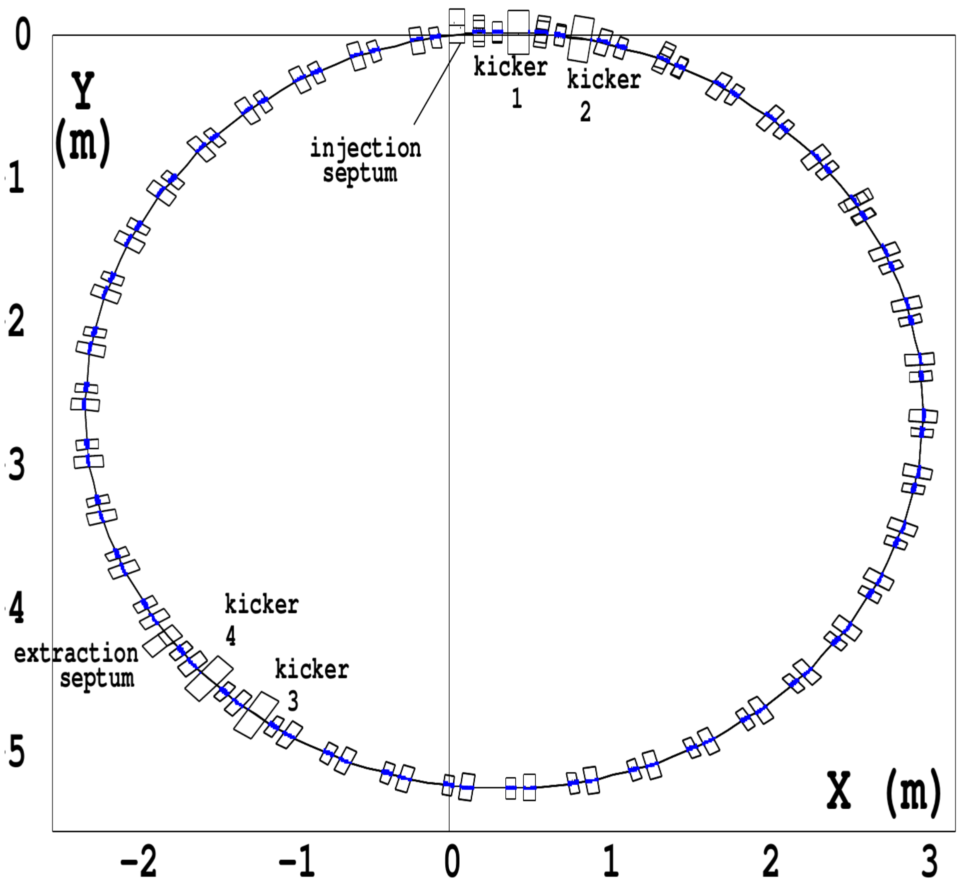
- However we will want to be able to inject multipole components, in order to simulate field defects.
So, we may want to consider 'MULTIPOL' as well

- **Following p. 8 (PART A o f the guide) :**

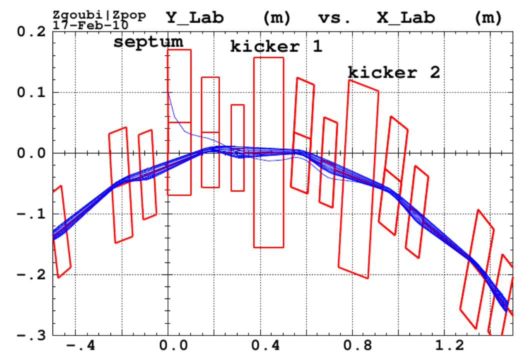
Let's jump to Zgoubi Users' Guide

QUADRUPO : pp. 138, 139,
MULTIPOL : p. 132

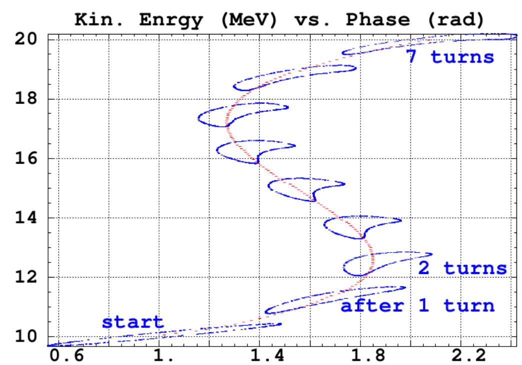
“How Sophisticated Can The Method be ?” - Cont’d. And Conclusion : Start-to-End in EMMA



EMMA ring in Zgoubi using the interface software “zpop”. The tracks of $14 - \frac{7}{19}$ accelerated turns are shown (blue), motion is clockwise.



A zoom on the injection region.



Serpentine motion from 10 to 20 MeV, 145 cavities crossed (7 full turns and an additional 12 cavities).