Recirculating Placet and

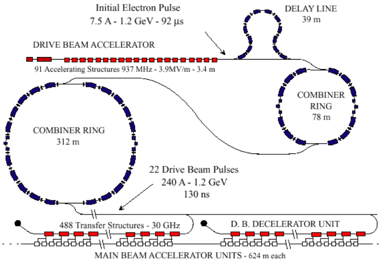
first insights on the Combiner Ring Length Measure

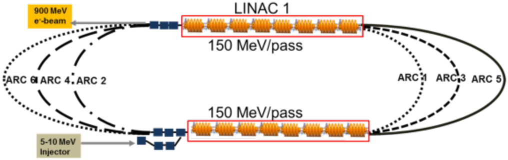
Dario Pellegrini (CERN, EPFL)

Jan 26, 2015

Recirculating Machines

- Recirculating machines are a special class of particle accelerators in which the beam circulates a few times in the same elements.
- The bunch train is often modified during operation (non-fixed bunch spacing and/or ordering).
- Neither linac nor ring codes are fully suited.
- Multibunch effects in these machines are typically handled with small, *ad hoc* codes.





Many different and complex topologies!

RPL: Recirculating Placet

Why a code designed to handle such machines?

Fundamental reason: compute the impact of multibunch effects with beam recirculation:

- *Long Range Wakefields - Higher Order Modes*,
- ion/electron clouds,

...on top of a full 6D tracking with a number of single particle/single bunch effects:

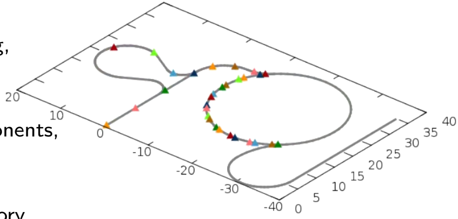
- *Incoherent* and Coherent *Synchrotron Radiation*,
- Short Range Wakefields and Impedances,

Corollary benefits:

- Clearer and easy-to-maintain scripts (no multiple definition of the same element, lattice unrolling, ...),
- Misalignments and timing errors correctly applied and propagated to the whole machine,

What does RPL ask you?

- Definition of one or more *injectors*:
  - create and store bunches for tracking,
  - define the initial beam structure.
- Definition of one or more *beamlines*:
  - standard, simple sequences of components,
  - include "beam instrumentation".
- Definition of one or more *dumps*:
  - delete the bunches, freeing the memory,
  - provide additional instrumentation.



- Connect together injectors, beamlines and dumps to create a *machine*:
  - complete description of the accelerator,
  - possible to adjust/edit properties after the previous definitions.

- Run the machine to obtain beam data!

What does RPL do for you?

- Tracks all the bunches determining their path in the machine at runtime,
- In case of many bunches, guarantees that each of them enters each beamline in the correct order,
- Handles all the time dependencies in the machine (element phases, HOMs damping, ...),
- Collects beam properties at the desired locations, bunch after bunch,
- Allows to follow a bunch along its path collecting many observables (Twiss, orbit, transport matrix/higher order tensors, losses, ...).

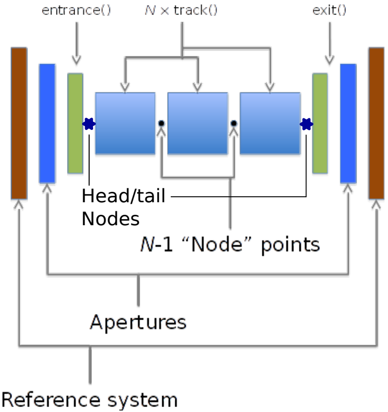
What is there behind the curtain?

RPL is a full 6D tracking code:

- entirely written in modern C++ from scratch,
- embedded into Placet featuring a SWIG-generated Tcl interface,
- designed to be easily expanded (add more beam models, more elements, more physics effects...),
- benefits A LOT from the experience maturated with the operation and maintaining of Placet,
- the interface is mostly compatible with PLACET, featuring a set of richer and more flexible new commands (e.g. beam creation, element slicing, machine topology definition, ...),
- in a good state of development, under constant expansion, already being tested in real-life cases (CTF3 measurements, LHeC design validation, ...).

Two insights on the RPL mechanisms

Structure of the element: an embedded integrator



Element as a collection of kicks:

- Misalignment/Aperture/Fringe fields;
- Sliced *thick* core (drift, dipole, quadrupole, RF cavity...);
- *Thin* kicks added between the slices to import physical effects (Radiation, wakefields, multipolar component, stray fields...).

Huge flexibility:

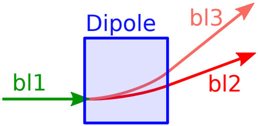
- Select which effects to add in each element;
- Split an effect over multiple kicks.

Generalised kicks: they do not necessary "kick" the beam but can:

- transform the beam (match Twiss, phase advance, generate offsets, ...);
- collect the required properties of the passing bunches.

Beamlines connections and bunch routing

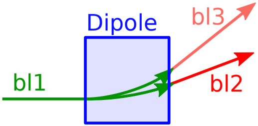
Example: dipole splitter



Routing according to energy:

- Beamline 1 ends at the dipole;
- Beamline 2 and 3 have two dipoles tuned at different energies.

```
Mac link -in bl1 -out bl2 -cmd {abs([bunch getd energy] - 20)}
Mac link -in bl1 -out bl3 -cmd {abs([bunch getd energy] - 10)}
```



Routing according to offset:

- Beamline 1 contains the dipole, bunches at different energies end up with different positions;
- Beamline 2 and 3 start after the dipole.

```
Mac link -in bl1 -out bl2 -cmd {abs([bunch getd cx] - 2e-3)}
Mac link -in bl1 -out bl3 -cmd {abs([bunch getd cx] - 1e-3)}
```

- The *bunch* keyword in the interface, becomes the actual bunch object being tracked by the C++ core: you can call its member functions (*getd* for instance)!
- The bunch proceeds in the beamline whose command evaluates to the *smaller* value.
- Beamlines merges are simpler: do not require to specify a routing criterion.

CTF3 Combiner Ring length measure

Motivation, description and preliminary results

Combiner Ring length measure: motivations

- The combiner ring adopts RF-injection to combine multiple trains of bunches in a single one with higher bunch frequency.
- The regularity of the recombined train influence the effectiveness of the deceleration.
- The *fractional length* of the combiner ring determines the final train structure.

- A wiggler allows to control the length increasing it, cannot make a ring shorter!
- We wanted to verify how to manipulate the ring length acting on the optics:
  - Dipole strengths,
  - Quadrupole strengths.

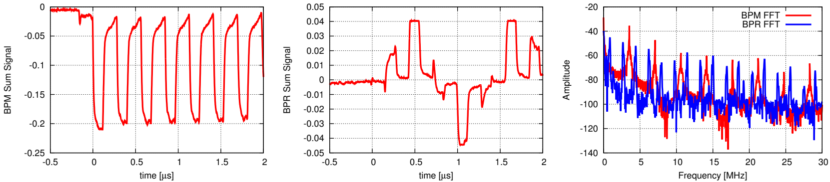
- Relatively easy measure using the signal of a BPR;
- Nice test bench for RPL.

The measurement methodology

- The BPR multiplies the signal coming from a button BPM for an internal 3 GHz sine signal.
- Turn after turn the phase slippage at 3 GHz becomes evident.
- Apply an FFT to the BPR signal: the revolution frequency splits into two sidebands
- The fractional length is directly obtained from the sidebands distance:

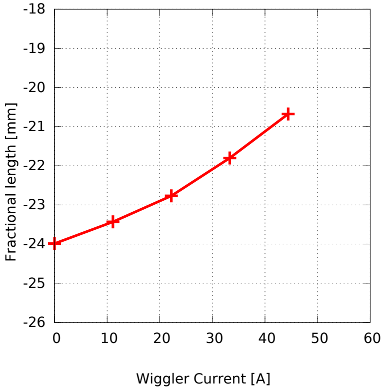
$$L_f = \pm \frac{d_s}{2 f_{rev}} \quad \left[\frac{c}{3\ \text{GHz}}\right]$$

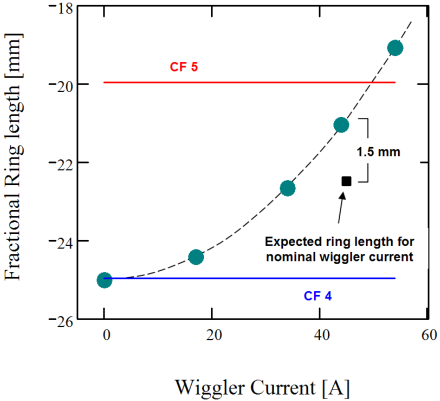
- The train fills half of the ring and is kept for many turns:



The effect of the wiggler

A scan of the wiggler allows to verify that the measure and the analysis are working:

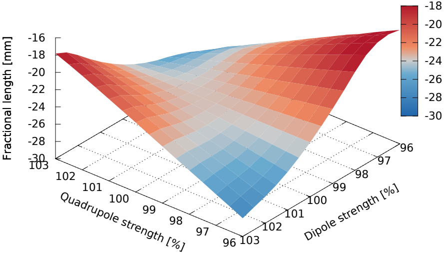




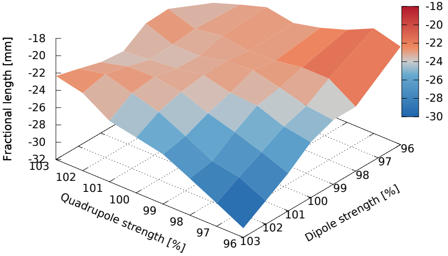
- The new measure (left) is in good agreement with the one made during the commissioning (right)
  - Several years of drifting;
  - Machine realignments.

Dipole and Quadrupole Scans

RPL Simulation:



Measure:

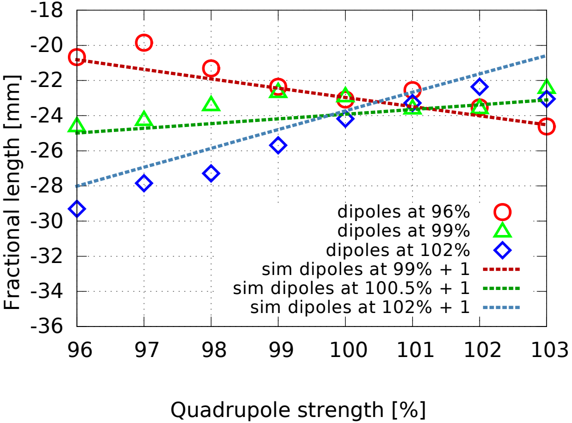


Losses are important:

- with an extreme scaling of the machine optics, part of the beam falls outside the aperture,
- beam tails are lost, the average beam energy changes, the measured length responds to it.

Track over a few turns to measure the length only with the transported part of the beam.

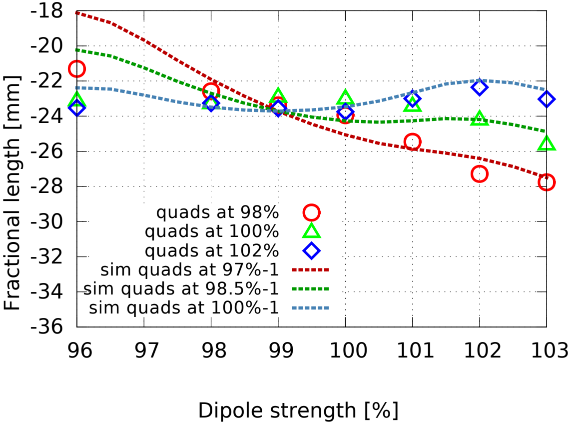
Constant Dipoles



The cutting of the simulated data is adjusted to fit the measures. Note the different dipole energies and the horizontal shift of the simulated data.

- Beam energy error?
- Issues with the power suppliers?

Constant Quadrupoles



Again the cutting of the simulated data is adjusted to fit the measures. Losses play a critical role here and give the hilly shape of the curves.

Conclusions

- Recirculating Placet or RPL is designed to compute multibunch tracking in recirculating lattices;
- It has a modern design and is getting productive;
- It can handle the CTF3 Combiner ring:
    - Optics and orbit computations (not shown here),
    - Single and multiturn 6D tracking with injection and extraction,
    - Computation of the vertical multibunch instability due to trapped modes in the old RF deflectors (not shown here).

- A measurement of the CR length as function of the dipole and quadrupole strengths has been performed;
- The possibility to decrease the ring length further than with the wiggler off has been demonstrated;
- Considering the losses due the chamber apertures the agreement with simulation has been improved.
- Need to understand the required adjustments together with the impact of:
    - Power suppliers
    - Magnet hysteresis
    - Energy error
    - Orbit effects