The Tracking Code RF-Track v2.5.3

Andrea Latina, CERN
andrea.latina@cern.ch

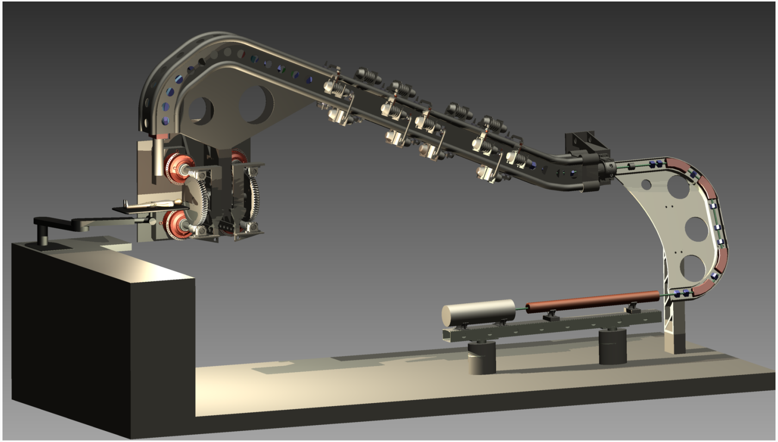
IHEP, Dongguan, China – 27 October 2025

Table of Contents

Introduction and highlights

Motivation for Developing RF-Track: the TULIP Project

A linac for hadron therapy featuring high-gradient S-band backward travelling-wave structures



S. Benedetti, A. Grudiev, and A. Latina, "High gradient linac for proton therapy", Phys. Rev. Accel. Beams 20, 040101 [2017]

What is RF-Track?

RF-Track Characteristics

- **RF-Track** is a CERN-developed particle tracking code
- Simulates **generation**, **acceleration**, and **transport** of **any** particle species
- Works with both **realistic 3D field maps** and **conventional beamline elements**

Applications

- **Photoinjectors** and **electron guns**
- **Positron sources**
- High-intensity **electron linacs**
- **RFQs** and high-intensity **proton and ion linacs**
- **Ion-electron cooling**
- **Inverse Compton scattering** X-ray sources
- **Ionisation cooling channel** of the **Muon Collider**

RF-Track Architecture

High-performance, Multi-Threaded C++ Core

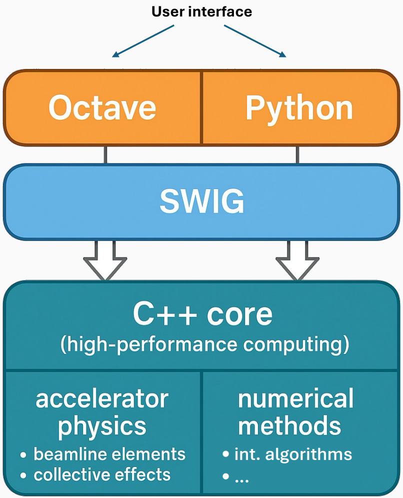
- Relativistic transformations
- Three- / Four-vectors
- 1D / 2D / 3D meshes
- Robust interpolators
- Quaternions
- Truncated Power Series Algebra (TPSA)

Accelerator Simulation Core

- Beamline elements
- Electromagnetic fields
- Collective effects

User Interface: Python / Octave

- Beamlines & beams
- Imperfections
- Orbit correction schemes



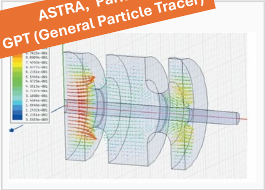
RF-Track Highlights

Key Capabilities

- Tracking in **time** and **space** of **any particle** at **any energy**, including **spin**
- **Realistic fields** and **arbitrary field maps**
- **Overlapping** elements
- **Space-charge effects**
- **Beam-beam effects**
- **Intra-beam scattering**
- **Wakefields** (short- and long-range)
- **Synchrotron radiation emission**
- **Multi-bunch Beam loading** in **SW** and **TW structures**
- **Particle-matter interaction**
- Arbitrary **element misalignments**
- **Beam-based alignment** algorithms
- **Back-tracking** in the presence of collective effects

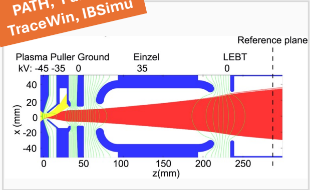
RF-Track Use Cases

ASTRA, Parmela,
GPT (General Particle Tracer)



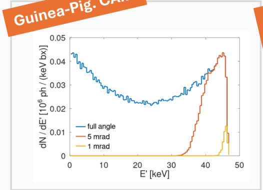
Photoinjectors and electron guns
(space-charge, IBS)

PATH, Parmila,
TraceWin, IBSimu



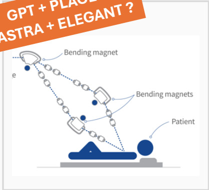
Positron, proton, and ion
sources & linacs

Guinea-Pig. CAIN



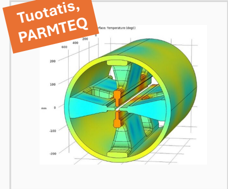
Inverse-Compton Scattering
Compact X-ray sources

GPT + PLACET ?
ASTRA + ELEGANT ?



Medical e⁻ linacs for
Flash therapy

Tuotatis,
PARMTEQ

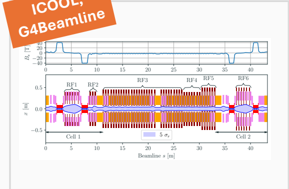


Radio-Frequency
Quadrupoles (RFQ)

PLACET,
ELEGANT

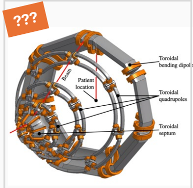
Linear Colliders, Free-Electron Lasers
(Wakefields, beam loading, …)

ICOOL,
G4Beamline



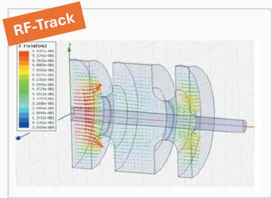
Muon Cooling
(Particle-matter interaction)

???

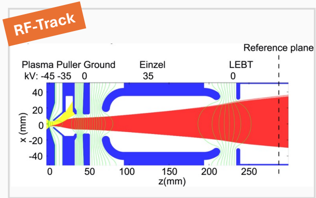


GaToroid
(Optics in complex fields)

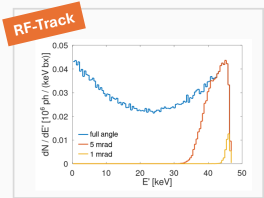
RF-Track Use Cases



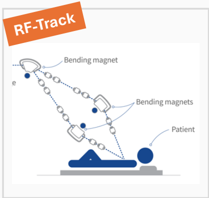
Photoinjectors and electron guns
(space-charge, IBS)



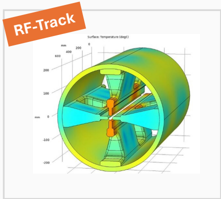
Positron, proton, and ion
sources & linacs



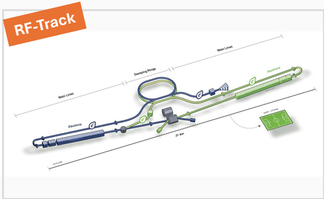
Inverse-Compton Scattering
Compact X-ray sources



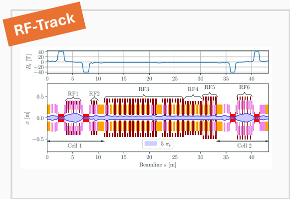
Medical e⁻ linacs for
Flash therapy



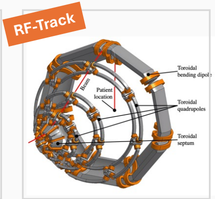
Radio-Frequency
Quadrupoles (RFQ)



Linear Colliders, Free-Electron Lasers
(Wakefields, beam loading, ...)



Muon Cooling
(Particle-matter interaction)



GaToroid
(Optics in complex fields)

Beam Models

Two Beam Models: Tracking in Space and in Time

RF-Track implements two beam models:

1. **Beam moving in space**: `Bunch6d()`
   - All particles have the same S position
   - The equations of motion are integrated in dS: $S \rightarrow S + dS$ (moves the bunch element by element)

$$(x \text{ [mm]},\ x' \text{ [mrad]},\ y \text{ [mm]},\ y' \text{ [mrad]},\ t \text{ [mm/c]},\ P \text{ [MeV/c]})$$

2. **Beam moving in time**: `Bunch6dT()`
   - All particles are considered at same time t
   - The equations of motion are integrated in dt: $t \rightarrow t + dt$
   - Particles can have $P_z < 0$ or even $P_z = 0$: particles can move backward

$$(X \text{ [mm]},\ P_x \text{ [MeV/c]},\ Y \text{ [mm]},\ P_y \text{ [MeV/c]},\ Z \text{ [mm]},\ P_z \text{ [MeV/c]})$$

Two Beam Models: Tracking in Space and in Time

RF-Track implements two beam models:

1. **Beam moving in space**: `Bunch6d()`
   - All particles have the same S position
   - The equations of motion are integrated in dS: $S \rightarrow S + dS$ (moves the bunch element by element)

$$(x \text{ [mm]},\ x' \text{ [mrad]},\ y \text{ [mm]},\ y' \text{ [mrad]},\ t \text{ [mm/c]},\ P \text{ [MeV/c]})$$

2. **Beam moving in time**: `Bunch6dT()`
   - All particles are considered at same time t
   - The equations of motion are integrated in dt: $t \rightarrow t + dt$
   - Particles can have $P_z < 0$ or even $P_z = 0$: particles can move backward

$$(X \text{ [mm]},\ P_x \text{ [MeV/c]},\ Y \text{ [mm]},\ P_y \text{ [MeV/c]},\ Z \text{ [mm]},\ P_z \text{ [MeV/c]})$$

For each macro particle also considers

$$\mathbf{m}: \text{ mass [MeV/c}^2\text{]},\quad \mathbf{Q}: \text{ charge } [e^+]$$

$$\mathbf{N}: \text{ nb of particles / macroparticle},\quad \mathbf{t_0}: \text{ creation time}^{(\star)} \qquad \boldsymbol{\tau}: \text{ lifetime } \textbf{[NEW!]}$$

$^{(\star)}$ only for beams moving in time.

RF-Track can simulate **multi-species beams** and the **creation** and **decay** of particles.

Bunch Creation

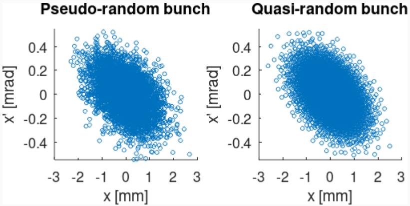
Particle bunches can be created in multiple ways:

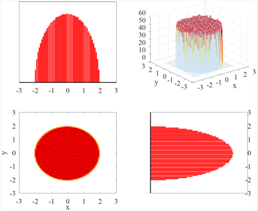
1. From a set of **Twiss parameters**
2. From **arbitrary distributions**, directly importing the phase space
3. From a **Photocathode** simulation similar to ASTRA's "Generator"
   [Credits: A. Aksoy (Ankara U.)]

   1D distributions: '**gaussian**', '**uniform**', '**plateau**', '**parabola**'

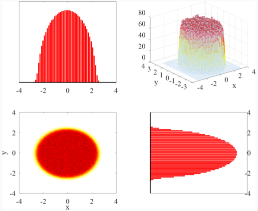
   2D distributions: '**radial-uniform**', '**radial-pleateau**', '**radial-gaussian**', '**radial-parabola**'

   3D distributions: '**ellipsoid**', '**isotropic**', '**fermi-dirac**'

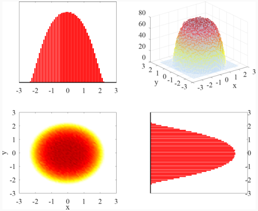




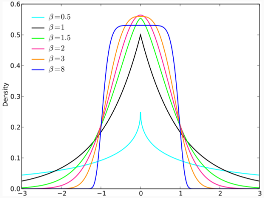
Radial-uniform



Radial-pleteau



Radial-parabola



Generalized Gaussian

Multi-Bunch Beams

Since version 2.3.0, it is possible to create **multi-bunch beams** with **arbitrary bunch type** and **spacing**.

Example of multi-bunch beam definition:

```
% Define a bunch
mass = electronmass;
charge = 1 * nC;
q = -1;
bunch = Bunch6d(mass, charge, q, phase_space);

% Define the train structure
num_of_bunches = 30; % train length
bunch_spacing = 1/3 * ns; % bunch spacing

% Define a beam
B0 = Beam(bunch, bunch_spacing, num_of_bunches);
```

A multi-bunch implementation eases and speeds up the computation of bunch-to-bunch collective effects.

Two Tracking Environments

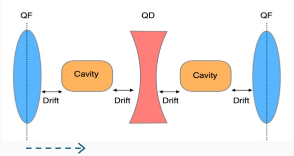
Lattice: *in-space* tracking

- A list of elements
- Tracks the beam element by element along the accelerator
- Elements can be arbitrarily misaligned

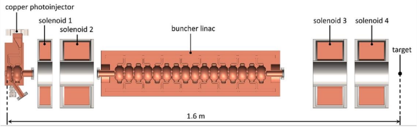
Volume: *in-time* tracking

- A **portion of 3D space**
- Elements can be **placed anywhere**
- **Arbitrary misalignment** via Euler angles (pitch, yaw, roll)
- Allows element **overlap**
- Allows **creation** of particles
- Can simulate **cathodes** and **field emission**
- Suitable for space-charge-dominated regimes

Example of Lattice: FODO cell

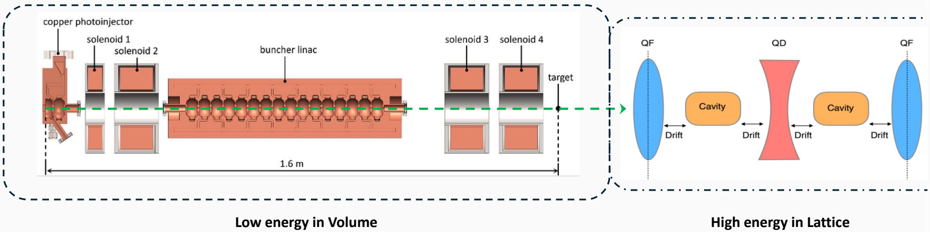


Example of Volume: Photoinjector



Lattice and Volume

Lattice and Volume can be used separately or in cascade. Example: a photoinjector and a linac

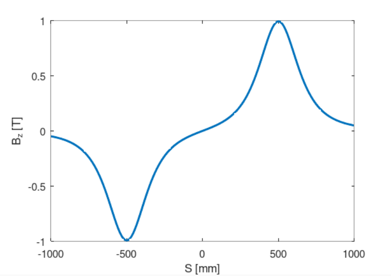


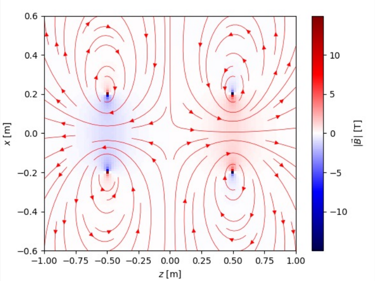
Volume (time integration) is typically preferred for **space-charge dominated regimes**, whereas **Lattice** (space integration) is more suitable for **ultra-relativistic regions** of the machine.

*Notice that Volume*s can be inserted in a Lattice, and *Lattice*s can be placed in a Volume.

Example of Volume

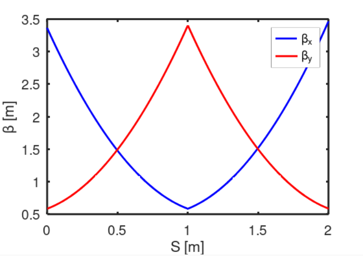
```
RF_Track;

L = 0;   % length [m]
B = 1;   % field at the center of the coil [T]
R = 0.2; % coil radius [m]

Cm = Coil(L, -B, R);
Cp = Coil(L, +B, R);

V = Volume();
V.add(Cm, 0, 0, -0.5, 'center');
V.add(Cp, 0, 0,  0.5, 'center');

figure(1)
Za = linspace(-1e3, +1e3, 1000); % mm
[E,B] = V.get_field(0, 0, Za, 0);

plot(Za, B(:,3));
xlabel('S [mm]');
ylabel('B_z [T]');
```

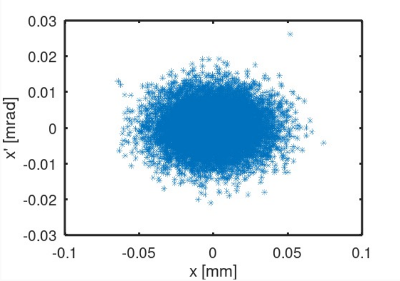




Example of Lattice

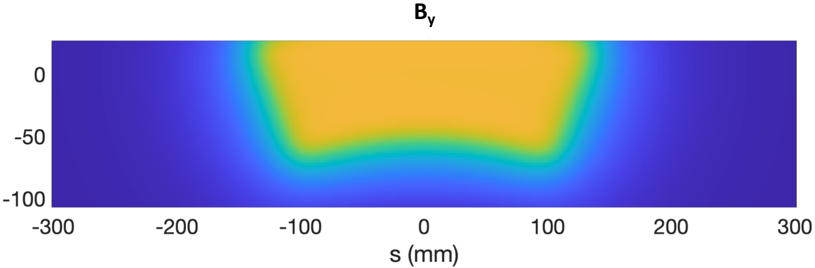
```
% load RF-Track
RF_Track;

% create a bunch from phase-space matrix
B0 = Bunch6d(electronmass, 200 * pC, -1, phase_space_matrix);

% create a lattice (1 FODO cell)
Lq = 0.4; % m
Ld = 0.6; % m
G = 1.2; % T/m

FODO = Lattice();
FODO.append (Quadrupole (Lq, G));
FODO.append (Drift (Ld));
FODO.append (Quadrupole (Lq, -G));
FODO.append (Drift (Ld));

% track the beam
B1 = FODO.track(B0);

% plot the phase space
T1 = B1.get_phase_space("%x %xp %y %yp");
scatter (T1(:,1), T1(:,2), "*");
xlabel ("x [mm]");
ylabel ("x' [mrad]");
```





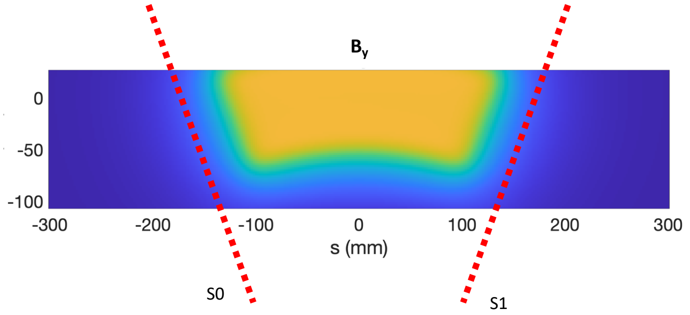
Boundaries of a Volume

Example of field map:



Boundaries of a Volume

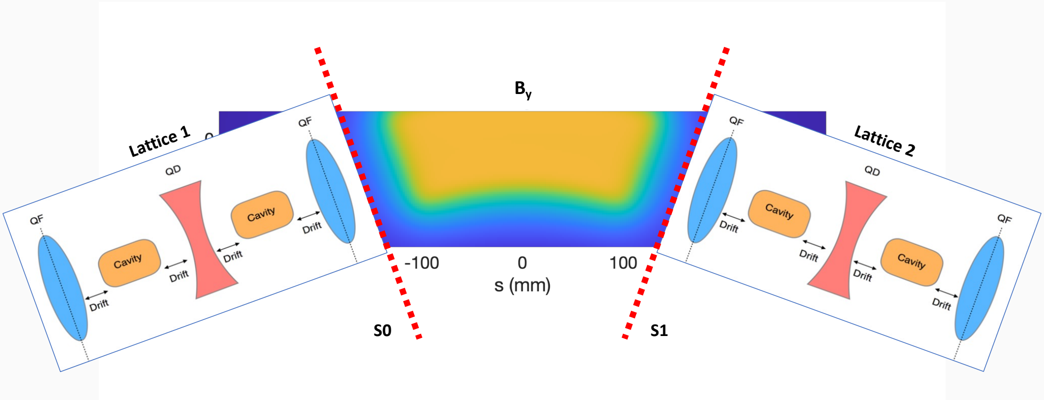
Boundaries of a Volume:



The boundaries of a Volume can have any orientation in space.

Volume as a Lattice Element

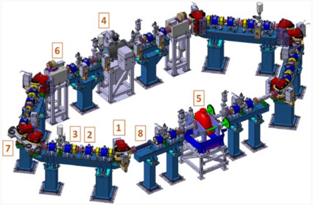
Boundaries of a Volume:



A Volume can be sandwiched between two Lattices.

The Case of Thom-X at IJCLab

RF-Track played a key role in the commissioning of Thom-X, the compact X-ray source based on Compton scattering at IJCLab.



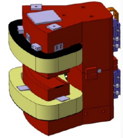
“Experiment on the table” with RF-track

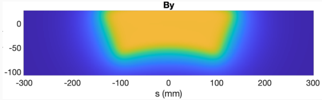
First study to measure ring frequency :

- Lattice with dipoles represented by **SBEND** (usual way) : F = 500.02 MHz
- Lattice with dipoles represented by **VOLUME** with realistic magnetic field : F =500.38 MHz, dispersive orbit. The same effect as in the experiment!

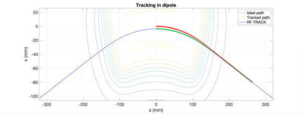
Short and small-radius dipoles, long fringe fields

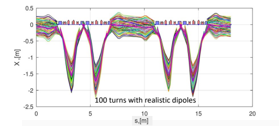
Features of dipoles	
Quantity	14 + 1 (pre-serie)
Radius of curvature	352 mm
Main field B_0	0.7 Tesla
Gap	42 mm
Good field region	+/- 20mm
Integral of field	184.59 mT.m
Current max.	275 Amp
Beam energy	from 50 to 70 MeV



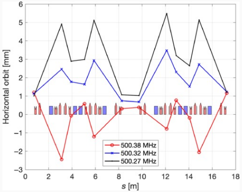


It was found that the beam trajectory in the dipoles is shorter wrt. to the ideal path => shorter pathlength and so smaller total circumference





Big step in understanding of the problem



Horizontal closed orbit measurements conducted during ring commissioning. The beam loss occurs below 500.27 MHz for the nominal optics configuration.

Phys. Rev. Accel. Beams **28**, 023401 – **Published 7 February, 2025**

Beamline Elements

Overview of the Beamline Elements

1. **Standard set** of **matrix-based symplectic** elements for Lattice:
   - **Sector bend, Quadrupole, Solenoid**
   - **Drift** (embedded in an optional constant electric and magnetic fields)

Overview of the Beamline Elements

1. **Standard set** of **matrix-based symplectic** elements for Lattice:
   - **Sector bend, Quadrupole, Solenoid**
   - **Drift** (embedded in an optional constant electric and magnetic fields)

2. **Field maps** (see next slides)

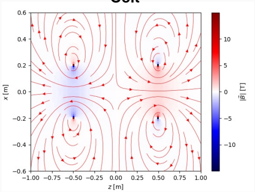
Overview of the Beamline Elements

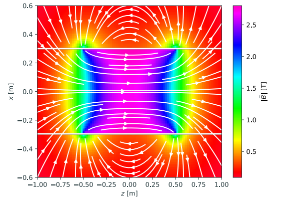
1. **Standard set** of **matrix-based symplectic** elements for Lattice:
   - **Sector bend, Quadrupole, Solenoid**
   - **Drift** (embedded in an optional constant electric and magnetic fields)

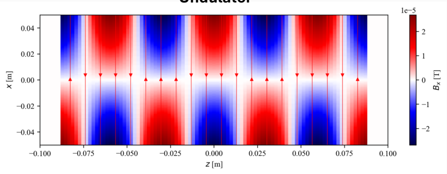
2. **Field maps** (see next slides)

3. **Special elements**:
   - **Absorber** (predefined materials: air, water, beryllium, lithium, tungsten, ...)
   - 3D analytic fields**: Coil** and **Solenoid**, **Undulator**, **Standing-wave** and **Traveling-wave** structures, **Adiabatic matching devices**, **Toroidal Harmonics**
   - **LaserBeam** for Inverse Compton Scattering simulations
   - **Electron Cooler**
   - **Transfer Line**: tracks through an arbitrary lattice given in form of Twiss table (phase advances, momentum compaction, 1^{st} and 2^{nd} order chromaticity are considered)
   - **Screens**: to capture the phase space at any point with any orientation in space

Elements Based on 3D Analytic Fields



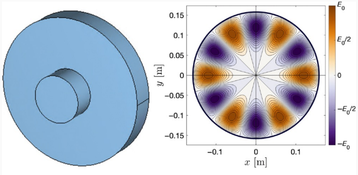




Coil

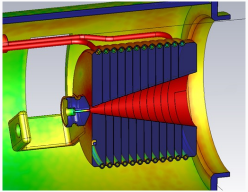
Solenoid

Undulator



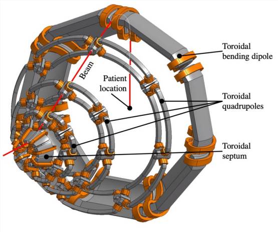
Traveling-, Standing-wave structure, TM_{nmp}

Pillbox cavity designed to support a 3 GHz TM_{610} mode



Adiabatic Matching Device

CLIC and FCCee positron sources



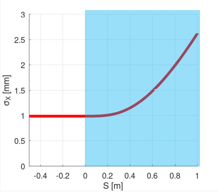
Toroidal Harmonics

GaToroid

Analytic fields are **exact**, **symplectic**, and **powerful** for design studies.

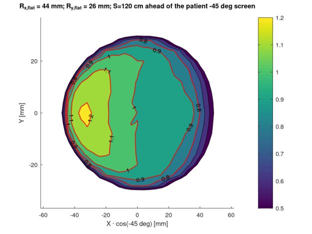
Special Elements

Absorber

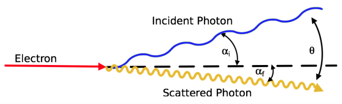


100 MeV electron beam traveling through air at CLEAR.

Screen

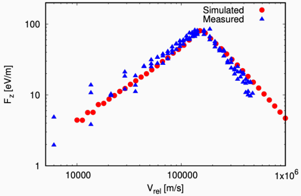


Laser Beam



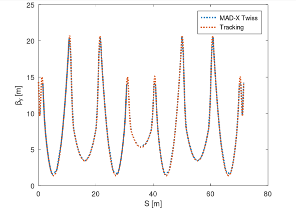
Inverse Compton Scattering

Electron Cooler



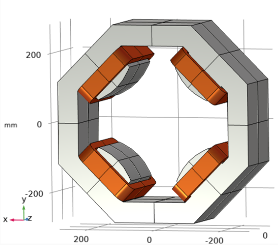
Cooling force experienced by a Pb beam in LEIR, as measured and simulated with RF-Track.

Transfer Line



Insert and match a transfer line in a single element

1D, 2D, 3D Field Maps



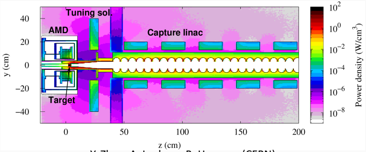
Static and oscillating RF fields

1D, 2D, and 3D Field Maps

RF-Track can import several types of oscillating RF field maps, which are interpolated *linearly* or *cubically*

- **1D field maps** (on-axis field)
  - It uses Maxwell's equations to reconstruct the 3D fields off-axis, assuming cylindrical symmetry
- **2D field maps**: given a field on a plane, applies cylindrical symmetry
- **3D field maps** of oscillating electro-magnetic fields
  - **Accepts 3D meshes of complex numbers**
  - **Accepts quarter field maps and automatically performs mirroring**
  - **For RF fields, allows specifying the input power supplied to the structure**
  - **Treats "NaN" in the field map as walls for accurate 3D loss maps** ⇒

Heat deposition in the FCC-ee positron capture area



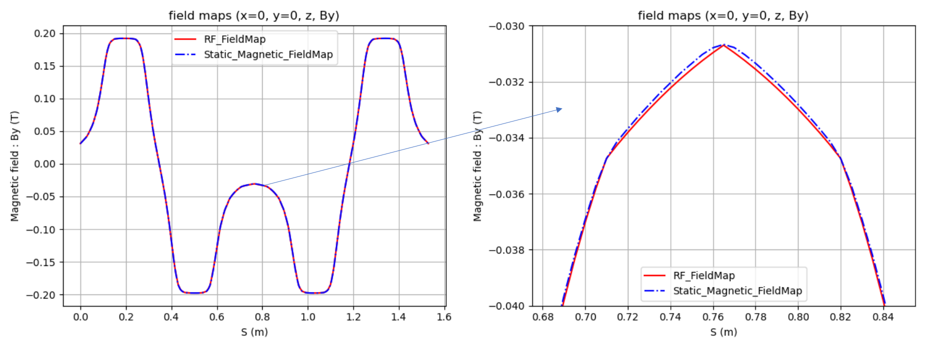
Y. Zhao, A. Lechner, B. Humann (CERN)

It also provides elements dedicated to **StaticElectric** and **StaticMagnetic** field maps

- They ensure curl-free (electric) and divergence-free (magnetic) interpolation of the field

Field Maps of Magnetic Elements

Static_Magnetic_FieldMap **corrects any input field map** (whether measured or computed), and makes it **physically correct**. This **ensures symplecticity**



Magnetic chicane for the FCC-ee's positron source.

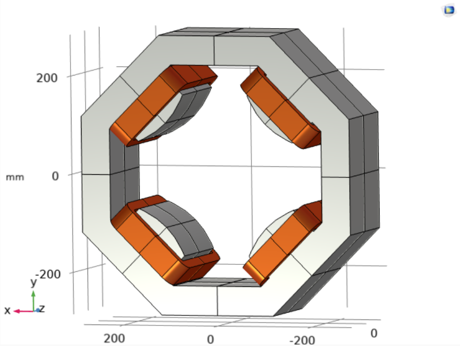
[Field map courtesy of Riccardo Zennaro (PSI); Plots courtesy of Yuting Wang (IJCLab)]

Long-term Tracking in Field Maps

As a test, we took the 3D field map of a quadrupole magnet.

We built a FODO cell and tracked a single particle through 1’000’000 cells.

- Quadrupole:
  - Strong fringe field
  - Field map: 41 x 41 x 121 points
- Integration algorithm: Symplectic Runge-Kutta 2nd order, “rk2”
- Initial phase space:
  - x = 1 mm
- A 100 MeV/c proton.



[Practical DEMONSTRATION.]

Credits:

- Quadrupole design and field map courtesy of Vera Korchevnyuk (CERN, EPFL)
- Tracking simulation scripts by Guglielmo Frisella (CERN, CNAO).

Integration Algorithms

In **field maps** and **analytic fields**, RF-Track integrates the equations of motion numerically:

- "**leapfrog**":
  - ⋆ fast, second-order accurate, symplectic
- “**analytic**” algorithm:
  - ⋆ integration assuming a locally-constant EM field
- **Higher-order, adaptive algorithms**:
  - ⋆"**rk2**" Runge-Kutta (2, 3)
  - ⋆"**rk4**" 4th order Runge-Kutta
  - ⋆"**rkf45**" Runge-Kutta-Fehlberg (4, 5)
  - ⋆"**rkck**" Runge-Kutta Cash-Karp (4, 5)
  - ⋆"**rk8pd**" Runge-Kutta Prince-Dormand (8, 9)
  - ⋆"**msadams**" multistep Adams in Nordsieck form (order varies dynamically between 1 and 12)

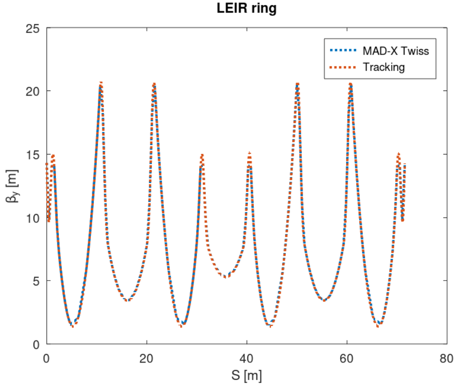
This applies to both **Lattice** and **Volume**. In both, **backtracking** is possible.

TransferLine

`TransferLine` allows the user to insert a full Twiss matrix as a single element.

```
% Input a MAD-X twiss file
R = TransferLine ('twiss_file.tfs');

% or a TWISS matrix
R = TransferLine (RING_TWISS, DQx, DQy, momentum_compaction, P_ref);
R.set_nsteps(100);
R.set_tt_nsteps(1000);
R.set_cfx_nsteps(100);
```

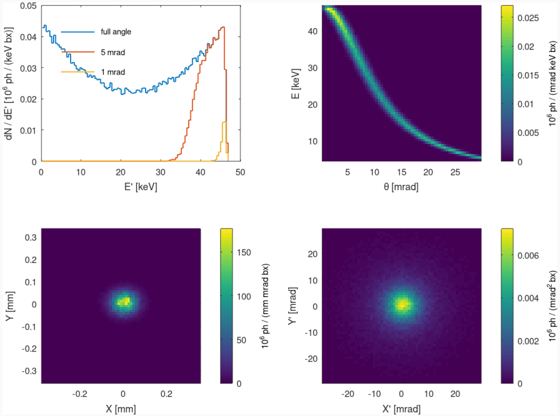
Collective effects can be distributed along the lattice.



The Element LaserBeam and Inverse Compton Scattering

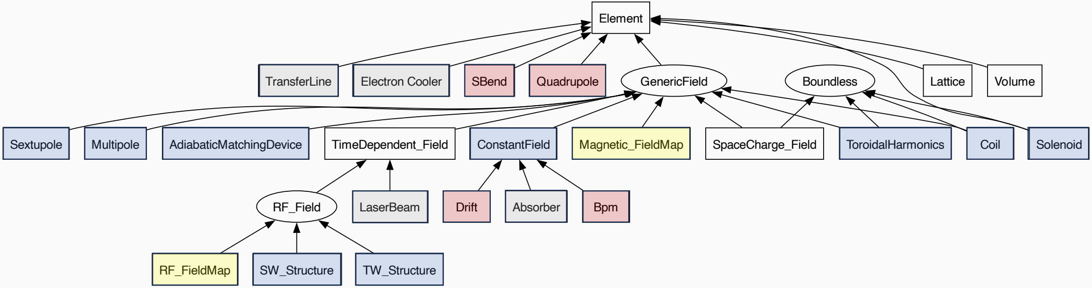
A collision with a **LaserBeam** can be added to any Lattice, using the element "LaserBeam":

```
X_angle = 2; % deg, crossing angle

nX = sind(180-X_angle);
nZ = cosd(180-X_angle);

%% Define laser-beam IP region
FP = LaserBeam(); % ICS interaction point
FP.pulse_energy = 28; % mJ, laser pulse energy
FP.pulse_length = 5; % ps, laser pulse length
FP.wavelength = 1030; % nm, laser wavelength
FP.set_direction (nX, 0, nZ); % laser incoming direction
FP.length = IP_length;
FP.set_IP_position (IP_length); % m
FP.R = 0.035; % mm, laser rms radius at waist, Gaussian profile
FP.M2 = 1.1; %
```



The **photons** are added to the bunch and **transported** through the beam line.

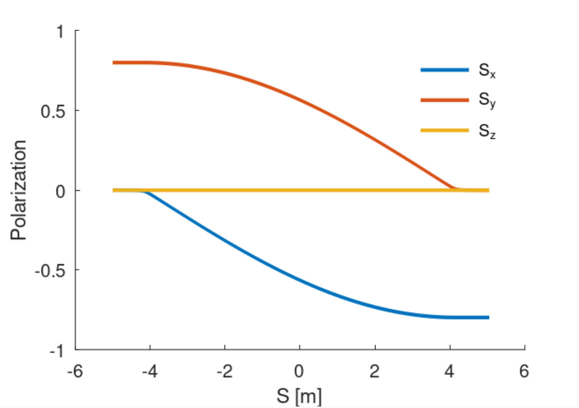
Element Hierarchy (to be updated...)



- Field maps 1D, 2D, 3D
- Analytic fields 3D
- Matrix based
- Special elements

Spin Tracking

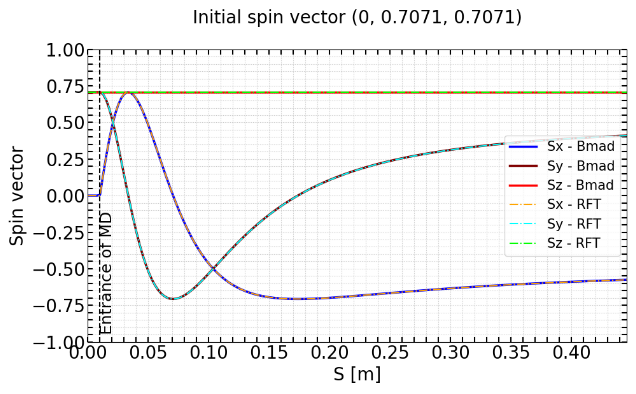
A **5 GeV electron beam** with **vertical polarization = 0.8**, through an **8.32 m long solenoid** with a $B_z = 3.15$ **T field:**



Solenoid of the ILC Spin Rotator

Spin Tracking - Benchmark Against BMAD

Evolution of **45 MeV** electron spin vector through the FCC-ee positron source's **HTS solenoid** (field map).



Courtesy of Yuting Wang (IJCLab)

User-Defined Elements

`UserElement` allows the user to customize RF-Track inserting a **Python object as an arbitrary element** in a Lattice.

Define myElement

```python
import RF_Track as rft
import numpy as np


class myElement(rft.UserElement):
    def track(self, bunch):

        print("PYTHON::Hi from Python!")
        print(f"PYTHON::bunch.S = {bunch.S} m\n")

        # Modify the phase space of the incoming bunch
        bunch[0].x += 5
        bunch[0].Pc = 50
```

Declare a Lattice

```python
# Create an instance of myElement
P = myElement()

# Crate a lattice with myElement in it
L = rft.Lattice()
L.append(P)
```

Track a bunch

```python
# Create a test bunch and track it
B0 = rft.Bunch6d(rft.electronmass, 0.0, -1, np.array([0.0, 0.0, 0.0, 0.0, 0.0, 100.0]))
B1 = L.track(B0)
```

Example of UserElement: RF-Track–BDSIM interface

An example application: The **interface with** BDSIM (v1.7.8), developed in collaboration with Stewart Boogert (Cockcroft Institute)

Example:

A **1 GeV electron beam against a 5cm long iron target** followed by a drift and and a rectangular bend

```
import RF_Track as rft
import bdsim
```

BDSIM Element

```
e = bdsim.Element()
e.name = "collimator"
e.type = bdsim.ElementType.RCOL
e.l = 0.05
e.xsize = 0.0
e.ysize = 0.0
e.material = "G4_Fe"
e = bdsim.rftrack.BDSIMElement(e);
```

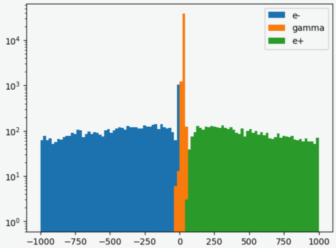
RF-Track Elements

```
d1 = rft.Drift(0.5)
d2 = rft.Drift(0.5)
b1 = rft.RBend(0.5, 0.02, P_over_q)
d3 = rft.Drift(1.0)
```

RF-Track Lattice+Tracking

```
lat = rft.Lattice()
lat.append(d1)
lat.append(e)
lat.append(d2)
lat.append(b1)
lat.append(d3)
b1 = lat.track(b0)
```

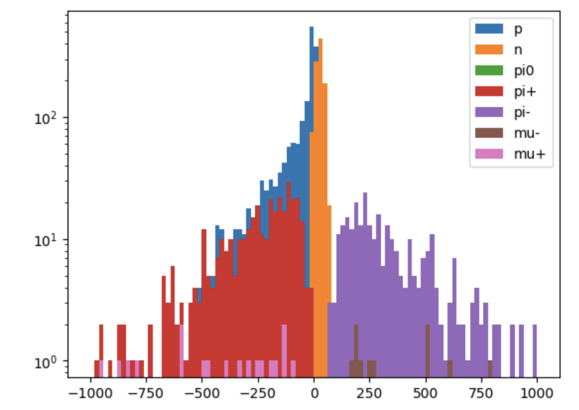
```
[7]: import matplotlib.pyplot as plt
     plt.hist(b1_df['x'][b1_df['Q'] == -1],100,(-1000,1000), label="e-");
     plt.hist(b1_df['x'][b1_df['Q'] == 0],100,(-1000,1000), label="gamma");
     plt.hist(b1_df['x'][b1_df['Q'] == 1],100,(-1000,1000), label="e+");
     plt.yscale('log')
     plt.legend()

[7]: &lt;matplotlib.legend.Legend at 0x3785d5d30&gt;
```



An **RF-Track electron bunch**, through a **BDSIM iron target**, **generated electrons**, **positrons** and **gammas** tracked in RF-Track.

Example of UserElement: RF-Track–BDSIM interface

A 5 GeV proton beam on a 80 cm long graphite target:

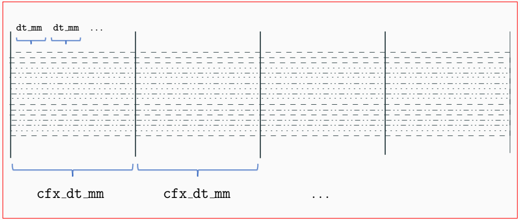


An **RF-Track proton bunch** on a **BDSIM graphite target**, followed by a **spectrometer in RF-Track**.

Collective and Single-Particle Effects

Tracking with Collective Effects

Staggered integration in **Volume:**



Volume

- **Fine Integration** of single-particle trajectories in '**dt_mm**', performed in parallel using high-order adaptive integration
- **Staggered Integration** of collective effects in **'cfx_dt_mm**', performed on the whole beam, using fixed-step-size leapfrog

Parallel tracking between consecutive kicks of collective effects. The process is analogous in **Lattice**, but the collective effects are associated to **each element**. Example:

```
E = Quadrupole( ... );
E.set_cfx_nsteps( 123 );
```

Overview of the Collective and Single-Particle Effects

Collective effects:

- **Space-charge**, full 3D, Particle-in-Cell (FFT) or P2P
  - Full computation of electric and magnetic effects
  - Beam-beam effects are automatically included
  - Optionally considers mirror charges at cathode
- **Short-range wakefields**:
  - Karl Bane's approximation
  - 1D user-defined spline, longitudinal monopole or transverse dipole
- Two models of **Long-range wakefields:**
  1. Sum of damped oscillators. Takes modes: frequency, amplitude, and Q factor
  2. 1D user-defined spline, longitudinal monopole or transverse dipole
- Self-consistent **Beam loading** effect in TW and SW structures
  - Given: R/Q, group velocity, and Q factors along the structure, computes the beam loaded fields
- **Intrabeam scattering**: hybrid-kinetic Monte Carlo method

Single-particle effects:

- **Incoherent Synchrotron Radiation** (from *any* fields)
- **Multiple Coulomb Scattering / Energy straggling / Energy loss**

Effects can be attached to **any element**, in **any number**.

Space-Charge Effects (1/4)

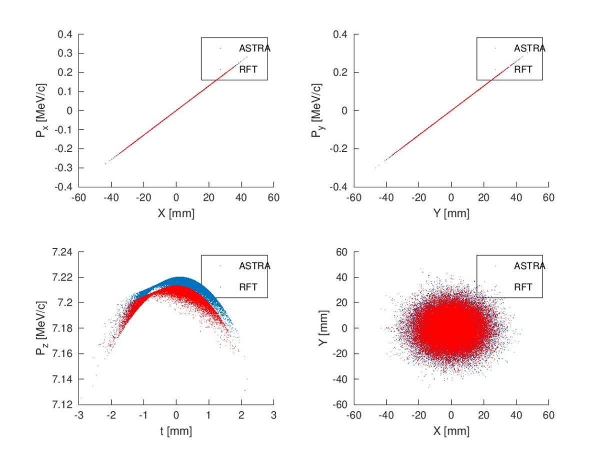
Benchmark against ASTRA:

Simulation of the **CLEAR photoinjector**:

- Q = 600 pC
- Gun, Ez = 100 MV/m, f = 3 GHz
- Peak energy phase for the reference particle
- Solenoid, Bz = 0.25 T   ON | **OFF**
- Space-charge   ON | **OFF**

50’000 macro particles

Solenoid **OFF** – Space-charge **OFF**



Space-Charge Effects (2/4)

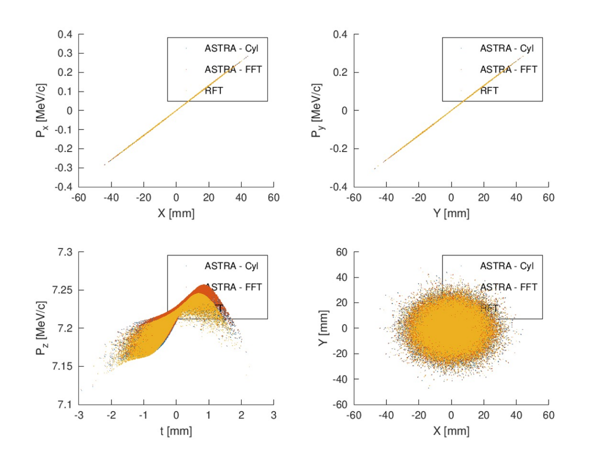
Benchmark against ASTRA:

Simulation of the **CLEAR photoinjector**:

- Q = 600 pC
- Gun, Ez = 100 MV/m, f = 3 GHz
- Peak energy phase for the reference particle
- Solenoid, Bz = 0.25 T   ON | **OFF**
- Space-charge   **ON** | OFF

50’000 macro particles

Solenoid **OFF** – Space-charge **ON**



Space-Charge Effects (3/4)

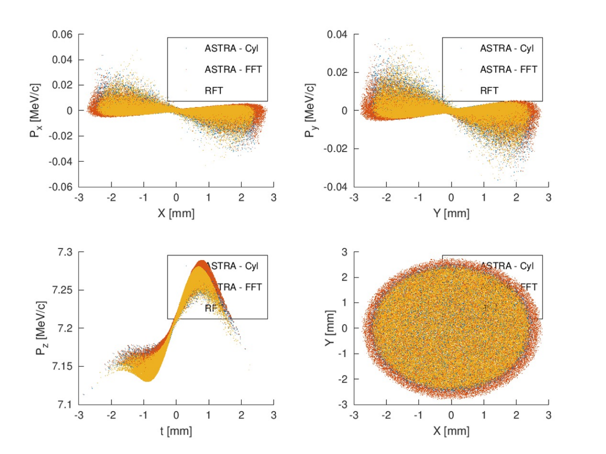
Benchmark against ASTRA:

Simulation of the **CLEAR photoinjector**:

- Q = 600 pC
- Gun, Ez = 100 MV/m, f = 3 GHz
- Peak energy phase for the reference particle
- Solenoid, Bz = 0.25 T **ON** | OFF
- Space-charge **ON** | OFF

50’000 macro particles

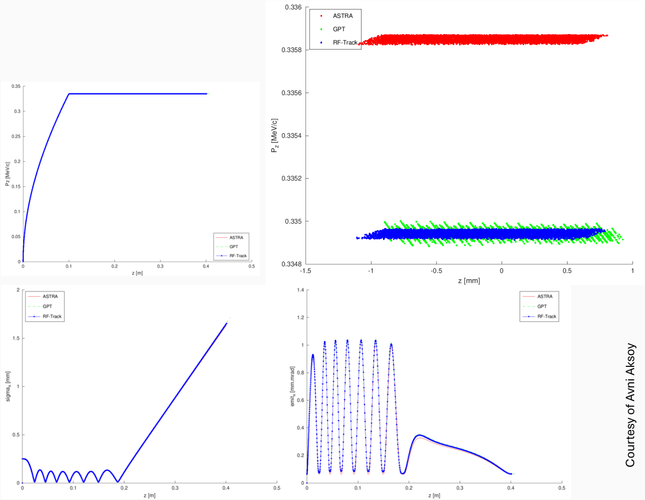
Solenoid **ON** – Space-charge **ON**



DC Gun

Electrostatic Gun (Astra, GPT, RFT)

Ez = -1 MV/m
B = realistic solenoid, B max = 0.25 T
No SC



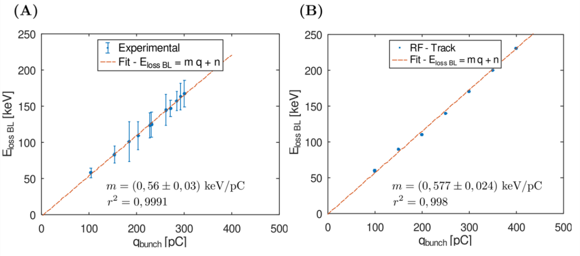
Courtesy of Avni Aksoy

Beam Loading in Standing-Wave Structures

Beam-loading effects have also been computed in standing-wave structures.

Follow the example of the photoinjector of the CLEAR test facility at CERN. The plots show the beam-induced energy loss of a train of 150 bunches with 1.5 GHz bunch-spacing, as a function of the bunch charge.

Beam parameter (end of linac)	Value range
Energy	60 - 220 MeV
Bunch charge	0.01 - 1.5 nC
Normalized emittances	3 um for 0.05 nC per bunch 20 um for 0.4 nC per bunch (in both planes)
Bunch length	~100 um - 1.2 mm
Relative energy spread	< 0.2 % rms (< 1 MeV FWHM)
Repetition rate	0.8 - 10 Hz
Number of micro-bunches in train	1 - 150
Micro-bunch spacing	1.5 or 3.0 GHz



(A) Experimental measurements
(B) RF-Track simulation

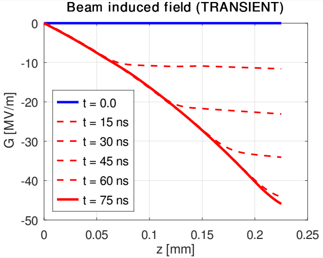
J. H. Olivares et al., "Implementation of the beam-loading effect in the tracking code RF-track based on a power-diffusive model", Frontiers in Physics, 2024

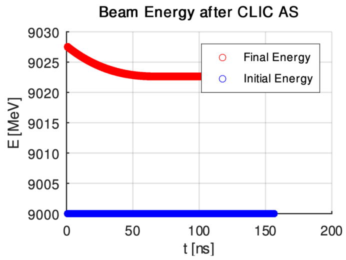
Beam Loading in Traveling-Wave Structures

A **power diffusive model** computes beam loaded field in an RF structure.
Both **transient** and **steady state** can be computed.
Beam-loading **compensation** schemes can be computed.

Usage example:

```
%% Transient Beam loading
BL = BeamLoading (Ncells, freq, phaseadvance, QQ, R_Q, VG);

%% RF-element from field map
TWS = RF_FieldMap_1d_CINT ( Ez, hz, L, freq, +1 );
TWS.set_odeint_algorithm( "rk2" );
TWS.set_P_map( Pmap );
TWS.set_P_actual( Pactual );

TWS.add_collective_effect ( BL );
TWS.set_cfx_nsteps ( 20 );
```





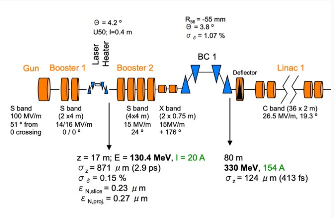
The plots show the effects of beam loading in a CLIC accelerator structure with a beam of 352 x 600 pC bunches with a 2 GHz bunch spacing.

Intra-beam Scattering – Experimental Validation

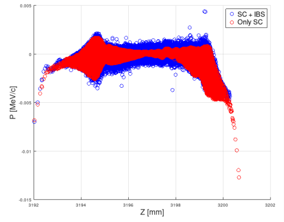
Implemented using a **Hybrid-Kinetic Monte Carlo method.**

First tests: **Slice-Energy Spread** (SES) in the **SwissFEL photoinjector** and **booster**: 200 pC bunch, from 0 to 330 MeV
- Z < 3 m: photocathode, 3 GHz electron gun, solenoid, up tp 7 MeV
- Z > 3 m: two 3 GHz Traveling-Wave structures - X-band linearizer

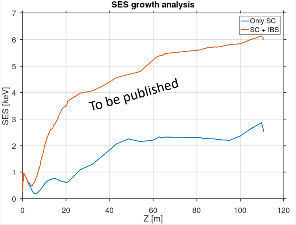
SwissFEL Injector and booster



SES at Z = 3 m



Linearized longitudinal phase space



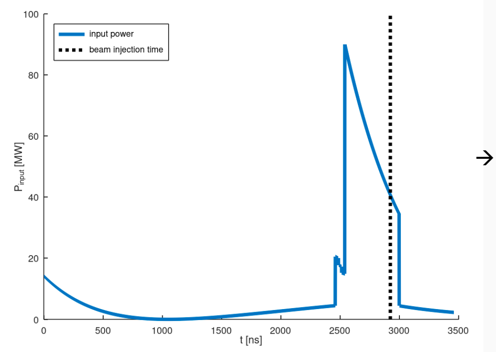
Measured SES = 6 keV

Matches with measurements.

[Work in progress for muon studies: Paula Desire Valdor (CERN, University of Groningen): SwissFEL studies: Thomas Lucas (PSI)]

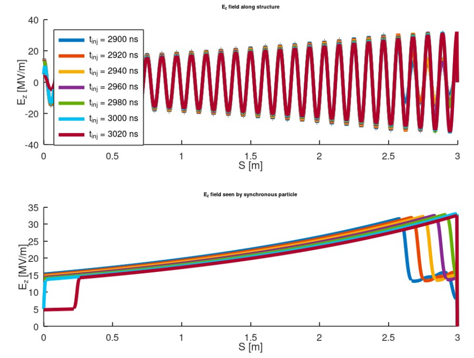
Beam Loading Compensation in the HE Linac

The element `TW_Field` was created to simulate **beam loading compensation** in the **High-Energy Linac** (to 20 GeV)



Modulated Input Power from the Klystron

→

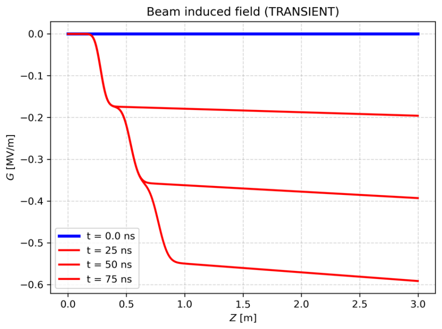


Field in the structure

[The modulated input power shown is the “Golden pulse”, A. Kurtulus, A. Grudiev (CERN)]

TW_Field

```
# Input power in watts
P = 60e6  # W

# Structure parameters
N_cells = 84
freq = 2.8e9 # Hz
ph = 2*np.pi/3 # phase advance

# Load Beam Loading parameters from text files
QQ  = np.loadtxt("Q_HE_linac.txt")                    # Quality factor
R_Q = np.loadtxt("r_Q_HE_linac.txt") * 1e3            # Ohm/m
VG  = np.loadtxt("Group_velocities_HE_linac.txt") / 100  # fraction of c

# Define the TW_Field object
TW = rft.TW_Field(P, QQ, R_Q, VG, freq, ph, N_cells)
TW = rft.TW_Field(P_vec, dt_power, t_injection, QQ, R_Q, VG, freq, ph, N_cells)
```



Beam loading initialization

```
BL = rft.BeamLoading (N_cells, freq, ph, QQ, R_Q, VG)

TW.add_collective_effect(BL) # Add transient BL to TW
TW.set_cfx_nsteps(10)        # 10 kicks per structure
```

TW_Field with **Golden Pulse** + **Beam Loading** for multi-bunch beam dynamics studies of **timing / energy errors**.

Benchmarks and Applications

Benchmarks and Example Applications

RF-Track is currently used for the design, optimisation, and simulation of:

- Medical applications (**DEFT facility**, collaboration CERN, CHUV, THERYQ), the **CLIC** and **FCC-ee positron sources** (CERN, IJCLab, PSI) and **FCC-ee pre-injector linacs** (CERN, PSI)

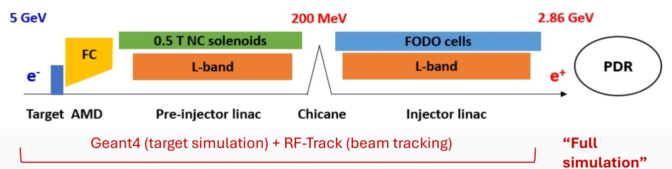
- **Linac4** (CERN), **Inverse-Compton Scattering sources** (CERN, IJCLab, INFN Ferrara, Korea University), and the **Cooling channel** of a future **Muon Collider** (CERN), etc.

I'll show two examples:

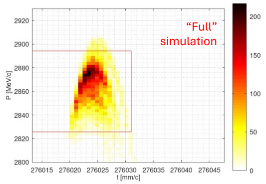
1. CLIC and FCC-ee positron sources
2. Cooling channel for a future muon collider
3. Linac4 H^- source
4. Linac4 RFQ and linac
5. ADAM's RFQ
6. Photoinjector for AWAKE
7. SuperKEKB Positron source

1. CLIC Positron Source

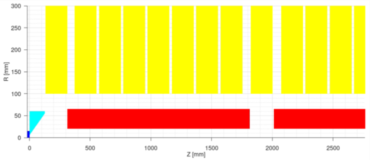
Start-to-end optimisation of the CLIC positron source



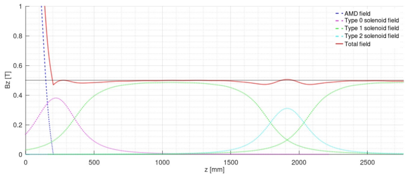
"Full simulation"



Longitudinal phase space @ 2.86 GeV



Schematic layout of solenoids



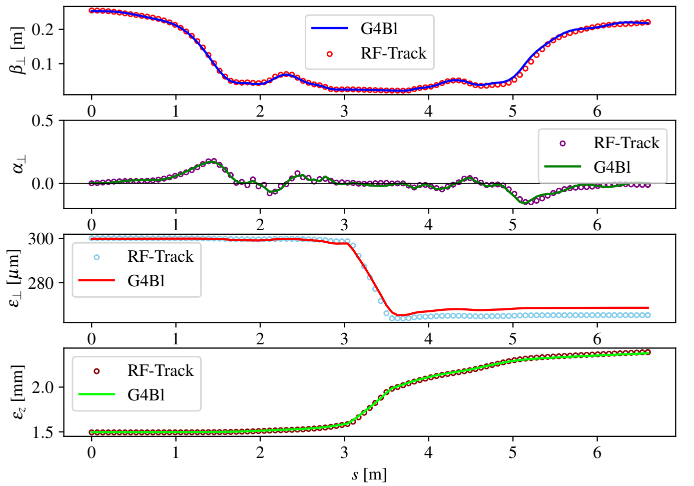
On-axis Bz of different components

- **More realistic simulations than any previous studies**
- **Start-to-end optimisation** with **higher positron yield than any previous studies ~1.8** @ 380 GeV (**~2.4** @ 3 TeV)

Work by **Yongke Zhao** (CERN)

2. Cooling Channel for a Future Muon Collider

Credits: Bernd Stechauner (CERN)



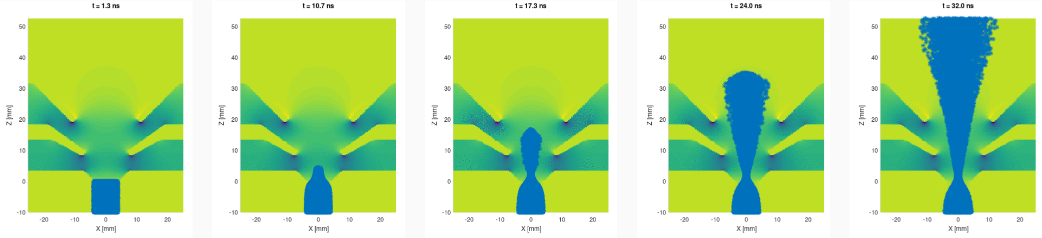
This simulation includes: 3D solenoids, standing-wave structures, absorbers – overlapping.

3. Linac4 at CERN: H^- Source

Preliminary results of the simulation of the Linac4 source: H^- ions are generated by ionising hydrogen gas, utilising plasma confinement, cesium injection for enhanced ion production, and electrostatic fields for extraction.

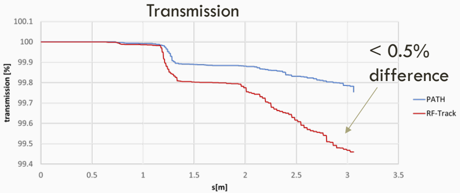
The ions are then accelerated in the Linac4 before being stripped of their electrons at 160 MeV to form protons.

- In this case, 5 eV H^- are simulated
- Simultaneous simulation of ions and electrons possible

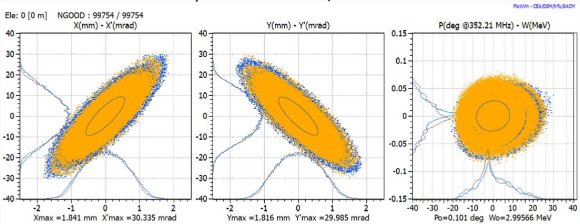


4. Linac4: RFQ

Work in progress with Giulia Bellodi (CERN)



Output 3 MeV beam, **PATH** vs **RF-Track**



4. Linac4

Work in progress with Giulia Bellodi (CERN)

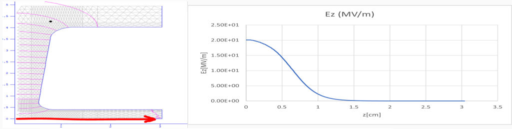
PATH: sequence of drifts and zero-length RF gaps giving acceleration kicks (thin gap model).

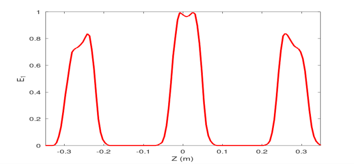
RF-Track: choice to use 1D field maps of on-axis longitudinal electric field obtained with Poisson-Superfish.

The longitudinal electric field is modelled with a generalized Gaussian function whose main parameters are fitted case by case to reproduce the different geometry and match the specified TTF (*ad-hoc* fitting procedure).

The same procedure is applied to all types of Linac4 cavities (DTL, CCDTL, PIMS): *77%* of the Linac is simulated using field maps.

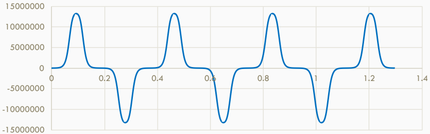
DTL-type gap





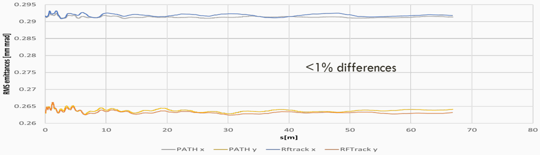
CCDTL tank

PIMS cavity

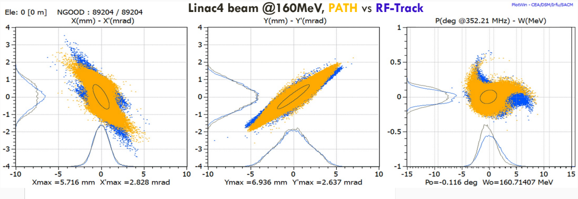


4. Linac4

Work in progress with Giulia Bellodi (CERN)



Linac4 with zero space charge



5. The RFQ of the ADAM Linear Accelerator for Proton Therapy

«LIGHT is a normal conducting 230 MeV medical proton linear accelerator being constructed by ADAM.

For the commissioning, RFQ beam dynamics simulations were performed with RF-Track by simulating the particles through the 3D field map.»

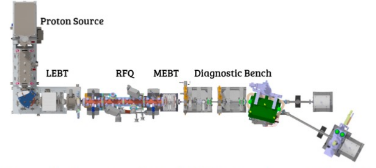


Figure 1: Layout of the LIGHT structures during the beam commissioning at 5 MeV.

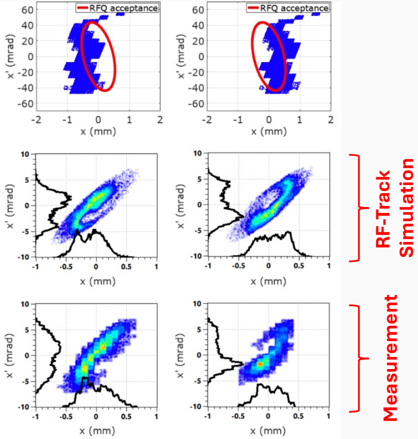
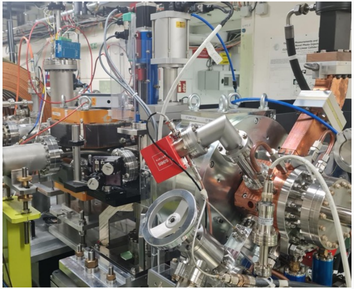
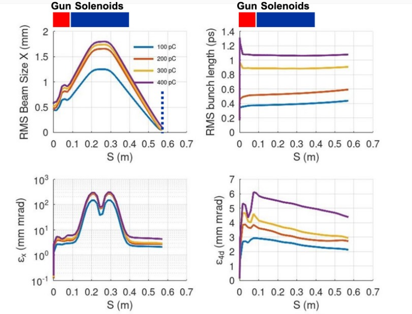


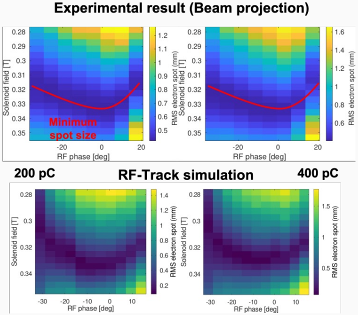
Figure 7: Horizontal phase space plots of the RFQ input beam when steered in the negative and positive x directions (first row), expected (second row) and the measured (third row) phase space plots after the RFQ for each case.

V. Dimov et al., "Beam commissioning of the 750 MHz proton RFQ for the LIGHT prototype", IPAC2018, Vancouver, BC, Canada, TUPAF002

6. Photoinjector for AWAKE

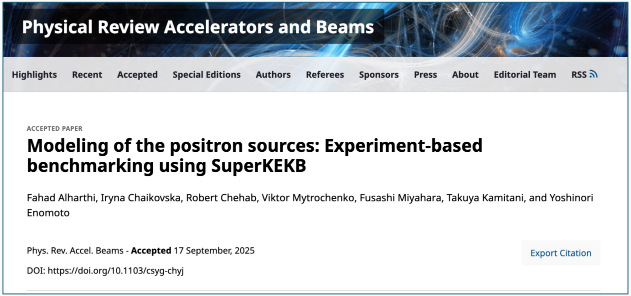






PhD Student **V. Musat** (CERN and Oxford University)

7. SuperKEKB Positron Source


Physical Review Accelerators and Beams

Highlights Recent Accepted Special Editions Authors Referees Sponsors Press About Editorial Team RSS

ACCEPTED PAPER

Modeling of the positron sources: Experiment-based benchmarking using SuperKEKB

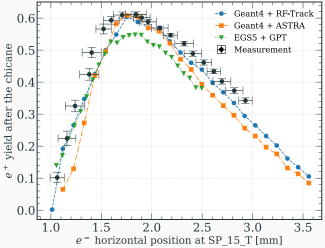
Fahad Alharthi, Iryna Chaikovska, Robert Chehab, Viktor Mytrochenko, Fusashi Miyahara, Takuya Kamitani, and Yoshinori Enomoto

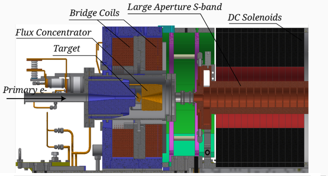
Phys. Rev. Accel. Beams - **Accepted** 17 September, 2025

DOI: https://doi.org/10.1103/csyg-chyj

Export Citation

e+ yield as a function of the primary e− beam impact position at the SP15T BPM

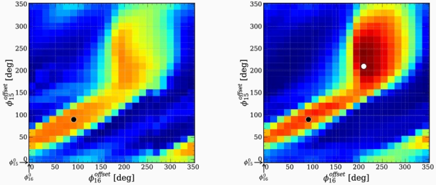




Target Area

Positron capture linac

RF phase scan of two klystron units KL15 and KL16 in the capture linac.



Measurement RF-Track simulation

Future Developments

- Coherent Synchrotron Radiation:
  - FCC-ee bunch and energy compressors
  - CLIC drive beam and main beam
- JSON input / output of lattices
  - Octave $\Longleftrightarrow$ Python exchange
  - RF-Track $\Longleftrightarrow$ Xsuite
- Recriculation (Lode Vanhecke, EPFL, CERN)
  - Replace PLACET2
  - Energy Recovery Linacs

RF-Track Community

Those I am aware of:

RFQs, hadron linacs: Linac4 (CERN), Peking University (China), Institute of Modern Physics, Lanzhou (China), CIEMAT (Spain), IFIC Valencia (Spain)

Photoinjectors: CLEAR (CERN), CTF2 (CERN), SwissFEL (PSI)

Positron sources: FCCee (IJClab, CERN), CLIC (CERN)

Optics in field maps: ThomX (IJClab), Synchrotron Soleil (France)

Inverse-Compton Scattering: X-band applications (CERN), ThomX (IJClab), Arizona State University (USA), LASA (Milan, Italy)

Medical accelerators: DEFT (CERN), PMB (France), THERYQ (France)

High-intensity electron linacs: CERN, Thales (France), IFIC Valencia (Spain), C^3 electron-positron collider (SLAC), EuSPARC (LNF-INFN)

Exotic scenarios: GaToroid (CERN, Oxford Univ.), Muon Cooling Channel (CERN)

RF-Track Resources

CERN website:

- Octave binaries, Documentation, Jupyter Notebooks Tutorials
  `https://gitlab.cern.ch/rf-track`

PyPi:

- Python version
  `https://pypi.org/project/RF-Track`

Reference Manual:

- `https://gitlab.cern.ch/rf-track/rf-track-reference-manual/-/raw/master/RF_Track_reference_manual.pdf?ref_type=heads`

Acknowledgments

The following people (in alphabetic order) **provided crucial contributions**

- **Avni Aksoy** (Ankara University, CERN, DESY)
  - Photo-injector simulation inspired by ASTRA's generator
- **Bernd Stechauner** (PhD student, CERN and University of Vienna)
  - Improved multiple Coulomb scattering implementation
- **Javier Olivares Herrador** (PhD student, CERN and University of Valencia)
  - Beam loading module
- **Lale Balat** (CERN Summer student and Boğazici University)
  - RF-Track test suite for automatic testing using Hierarchical Data Format (HDF5)
- **Paula Desire Valdor** (PhD student, CERN and Groningen University)
  - Intra-beam scattering module

Summary

- **RF-Track** is a **CERN-developed** particle tracking code designed to simulate the **generation**, **acceleration**, and **transport** of **any particle species** through an accelerator. It supports both realistic **3D field maps** and **conventional beamline elements**.
- RF-Track allows tracking **in both time and space** domains and includes a comprehensive set of **single-particle** and **collective effects**, including:
  - **Space-charge, Beam–beam interactions, Beam loading** in **standing- and traveling-wave structures, Short-** and **long-range wakefields, Synchrotron radiation emission, Multiple Coulomb scattering,** and **Intra-beam scattering**
- These capabilities make RF-Track ideally suited for the simulation of **high-intensity accelerator systems**. It has been successfully applied to:
  - **Electron linacs** including for medical applications and **FCCee Injectors**,
  - **Photoinjectors,**
  - **Positron sources, including the FCCee positron source**
  - **Proton linacs,**
  - **Ion–electron cooling**, and
  - **Inverse Compton scattering** X-ray sources,
  - **The ionisation cooling channel** of a future **muon collider**

Thank you for your attention!

Gitlab

PyPi