ECloud Simulations in the Crab Cavities

L. Giacomel, G. Iadarola, J.-L. Vay

November 10, 2020

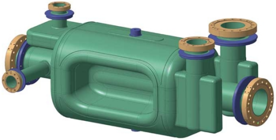
Many thanks to: G. Arduini, R. Calaga, M. Carlà, S. De Santis, M. Furman, D. Grote, T. Luo, E. Metral, K. Paraschou, G. Penn, L. Sabato, B. Salvant, M. Schwinzerl, C. Zannini

Outline

ECloud in the Crab Cavities

Three contributions to the electromagnetic (EM) field in the cavity:

- RF mode
- beam-induced field
- self-fields of the ECloud



Each contribution must be computed appropriately.

PyECLOUD Is Not Enough

PyECLOUD implements 2D quasi-static Particle-In-Cell (PIC). For the Crab cavities this is not enough because:

PyECLOUD Is Not Enough

PyECLOUD implements 2D quasi-static Particle-In-Cell (PIC). For the Crab cavities this is not enough because:

- 2D simulations would miss the possible **longitudinal motion** of the ECloud

PyECLOUD Is Not Enough

PyECLOUD implements 2D quasi-static Particle-In-Cell (PIC). For the Crab cavities this is not enough because:

- 2D simulations would miss the possible **longitudinal motion** of the ECloud
- the rapid motion of electrons pushed by the cavity might require an **electromagnetic solver**

PyECLOUD Is Not Enough

PyECLOUD implements 2D quasi-static Particle-In-Cell (PIC). For the Crab cavities this is not enough because:

- 2D simulations would miss the possible **longitudinal motion** of the ECloud
- the rapid motion of electrons pushed by the cavity might require an **electromagnetic solver**
- an electromagnetic solver is needed to compute the **cavity fields**

PyECLOUD Is Not Enough

PyECLOUD implements 2D quasi-static Particle-In-Cell (PIC). For the Crab cavities this is not enough because:

- 2D simulations would miss the possible **longitudinal motion** of the ECloud
- the rapid motion of electrons pushed by the cavity might require an **electromagnetic solver**
- an electromagnetic solver is needed to compute the **cavity fields**

Warp, a 3D PIC framework developed at LBNL, offers many of the required features.

PyECLOUD Is Not Enough

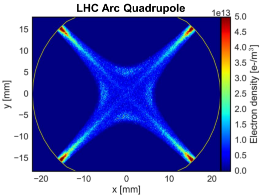
PyECLOUD implements 2D quasi-static Particle-In-Cell (PIC). For the Crab cavities this is not enough because:

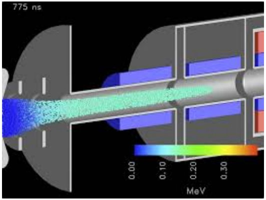
- 2D simulations would miss the possible **longitudinal motion** of the ECloud
- the rapid motion of electrons pushed by the cavity might require an **electromagnetic solver**
- an electromagnetic solver is needed to compute the **cavity fields**

Warp, a 3D PIC framework developed at LBNL, offers many of the required features.

We started a collaboration with the Warp developers and I spent time in Berkeley to learn and develop the required simulation tools.

PyECLOUD and Warp

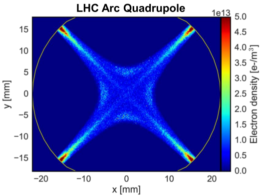


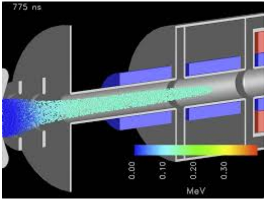


PyECLOUD

Warp

PyECLOUD and Warp





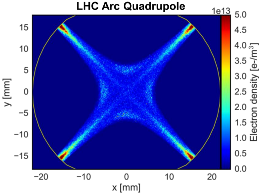
PyECLOUD

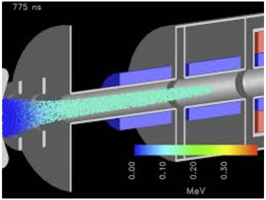
- 2D PIC code

Warp

- 3D PIC code

PyECLOUD and Warp





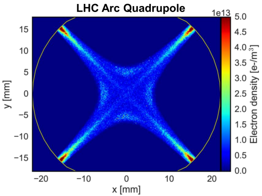
PyECLOUD

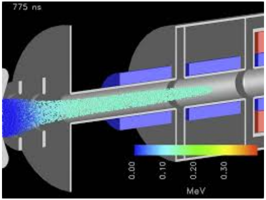
- 2D PIC code
- Quasistatic

Warp

- 3D PIC code
- Quasistatic or electromagnetic

PyECLOUD and Warp





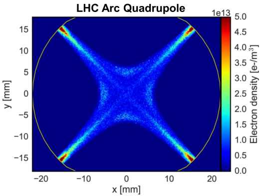
PyECLOUD

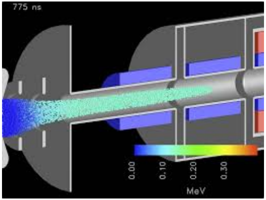
- 2D PIC code
- Quasistatic
- Implements several secondary emission models

Warp

- 3D PIC code
- Quasistatic or electromagnetic
- Doesn't implement secondary emission

PyECLOUD and Warp





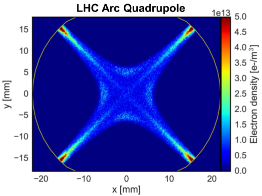
PyECLOUD

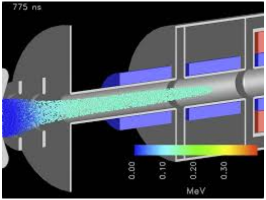
- 2D PIC code
- Quasistatic
- Implements several secondary emission models
- Code used for beam-induced multipacting simulations in the LHC complex

Warp

- 3D PIC code
- Quasistatic or electromagnetic
- Doesn't implement secondary emission
- Recently mainly used for laser-plasma acceleration simulations

PyECLOUD and Warp





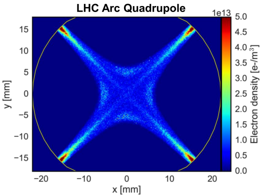
PyECLOUD

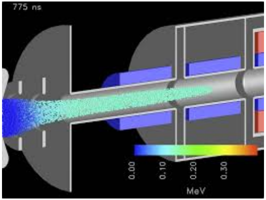
- 2D PIC code
- Quasistatic
- Implements several secondary emission models
- Code used for beam-induced multipacting simulations in the LHC complex

Warp

- 3D PIC code
- Quasistatic or electromagnetic
- Doesn't implement secondary emission
- Recently mainly used for laser-plasma acceleration simulations
- Used in the past for ECloud simulations (in coupling with POSINST)

PyECLOUD and Warp





PyECLOUD

- 2D PIC code
- Quasistatic
- Implements several secondary emission models
- Code used for beam-induced multipacting simulations in the LHC complex

Warp

- 3D PIC code
- Quasistatic or electromagnetic
- Doesn't implement secondary emission
- Recently mainly used for laser-plasma acceleration simulations
- Used in the past for ECloud simulations (in coupling with POSINST)

POSINST is an old package for ECloud simulations. It is equivalent to PyECLOUD, but it's written in Fortran (the code is not very readable), its coupling with Warp is unstable and its future is unclear.

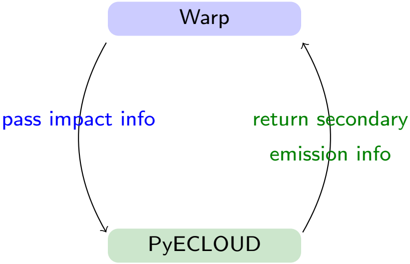
We decided to replace it with PyECLOUD.

Outline

Warp-PyECLOUD

Warp has a Python wrapper from which we can call routines belonging to other Python packages.

We decided to exploit this feature to interface Warp with PyECLOUD.

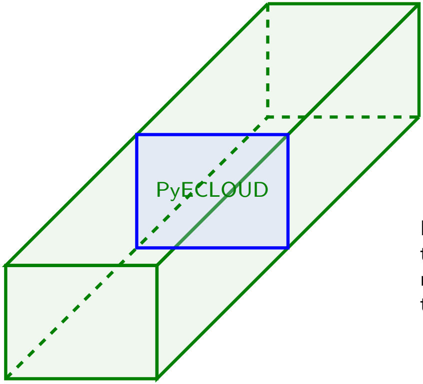


Details can be found in the presentation "Development of WARP simulations for 3D RF structures" given at the Electron Cloud Meeting #73

The interface has been developed by L. Giacomel, G. Iadarola, J-L Vay during Gianni's visit at LBNL in October 2019.

Warp-PyECLOUD benchmark

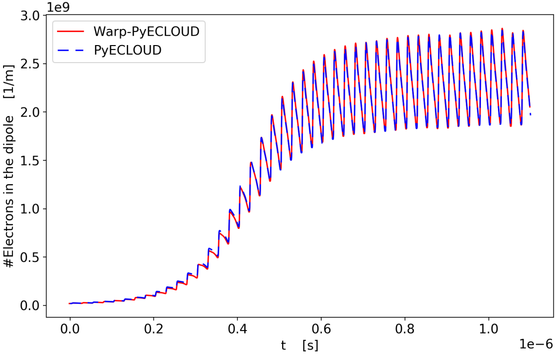
We use as a benchmark case a one-meter-long dipole with rectangular beam pipe, to validate the newly developed code.



In this case, we can reasonably compare the PyECLOUD 2D simulation with the middle section of the Warp 3D simulation.

Warp-PyECLOUD Benchmark

We compare the number of electrons per meter of dipole.



Runtimes:

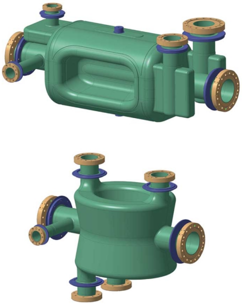
- Warp-PyECLOUD: 12.5h (MPI-parallelized on a 24-core machine)
- PyECLOUD: 20 minutes (serially)

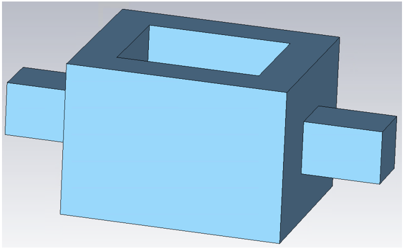
Outline

Simplification of the Crab Cavities

We initially simplified the structure of the Crab Cavities in order to avoid the numerical artifacts given by the staircasing approximations.

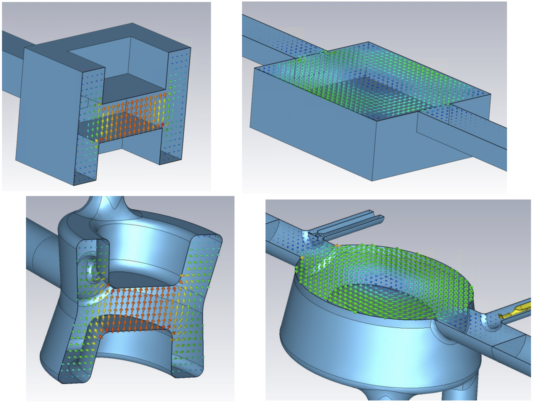
This has been useful to simply study the dynamics in the cavities and the properties of the different solvers.





Non-self-consistent Simulations

To begin with, we carried out preliminary simulations by computing the RF mode with the eigenmode solver by CST and importing it into Warp.



Non-self-consistent Simulations - Crabbing

We proved that the squared cavity can actually crab proton bunches and we compared it to a thin lens model in which the protons see a vertical kick:

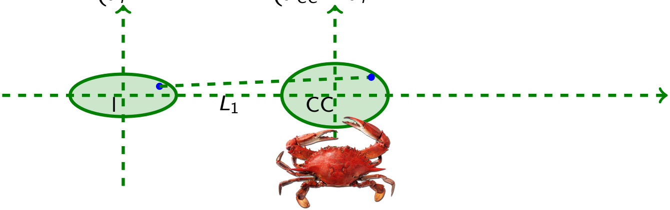
$$\Delta y'_{CC}(z_{CC}) = \frac{V_t}{E_0} sin\left(\frac{2\pi f_{CC}}{c} z_{CC} + \varphi_{CC}\right)$$

Non-self-consistent Simulations - Crabbing

We proved that the squared cavity can actually crab proton bunches and we compared it to a thin lens model in which the protons see a vertical kick:

$$\Delta y'_{CC}(z_{CC}) = \frac{V_t}{E_0} sin\left(\frac{2\pi f_{CC}}{c} z_{CC} + \varphi_{CC}\right)$$

$$\begin{cases} y_I \\ y'_I \end{cases} \qquad \begin{cases} y_{CC} = y_I + L_1 y'_I \\ y'_{CC} = y'_I \end{cases}$$

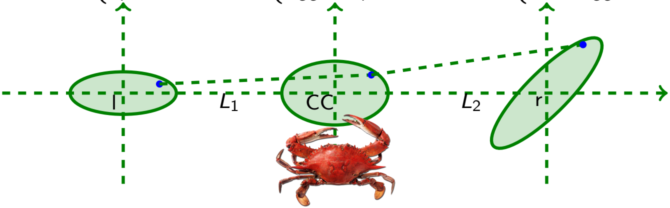


Non-self-consistent Simulations - Crabbing

We proved that the squared cavity can actually crab proton bunches and we compared it to a thin lens model in which the protons see a vertical kick:

$$\Delta y'_{CC}(z_{CC}) = \frac{V_t}{E_0} sin\left(\frac{2\pi f_{CC}}{c} z_{CC} + \varphi_{CC}\right)$$

$$\begin{cases} y_l \\ y'_l \end{cases} \qquad \begin{cases} y_{CC} = y_l + L_1 y'_l \\ y'_{CC} = y'_l \end{cases} \qquad \begin{cases} y_r = y_{CC} + L_2(y'_{CC} + \Delta y'_{CC}) \\ y'_r = y'_{CC} + \Delta y'_{CC} \end{cases}$$

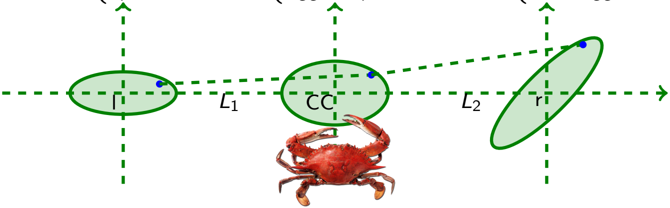


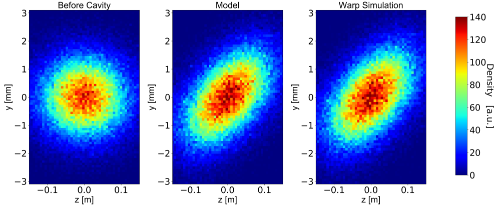
Non-self-consistent Simulations - Crabbing

We proved that the squared cavity can actually crab proton bunches and we compared it to a thin lens model in which the protons see a vertical kick:

$$\Delta y'_{CC}(z_{CC}) = \frac{V_t}{E_0} sin\left(\frac{2\pi f_{CC}}{c} z_{CC} + \varphi_{CC}\right)$$

$$\begin{cases} y_l \\ y'_l \end{cases} \qquad \begin{cases} y_{CC} = y_l + L_1 y'_l \\ y'_{CC} = y'_l \end{cases} \qquad \begin{cases} y_r = y_{CC} + L_2(y'_{CC} + \Delta y'_{CC}) \\ y'_r = y'_{CC} + \Delta y'_{CC} \end{cases}$$



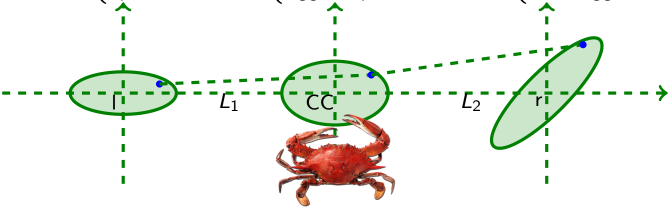


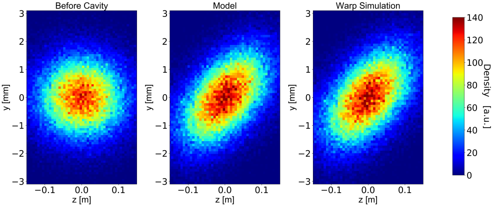
Non-self-consistent Simulations - Crabbing

We proved that the squared cavity can actually crab proton bunches and we compared it to a thin lens model in which the protons see a vertical kick:

$$\Delta y'_{CC}(z_{CC}) = \frac{V_t}{E_0} sin\left(\frac{2\pi f_{CC}}{c} z_{CC} + \varphi_{CC}\right)$$

$$\begin{cases} y_l \\ y'_l \end{cases} \quad \begin{cases} y_{CC} = y_l + L_1 y'_l \\ y'_{CC} = y'_l \end{cases} \quad \begin{cases} y_r = y_{CC} + L_2(y'_{CC} + \Delta y'_{CC}) \\ y'_r = y'_{CC} + \Delta y'_{CC} \end{cases}$$





More details can be found in the presentation "Development of WARP simulations for 3D RF structures", Electron Cloud Meeting #73

Different Simulation Approaches

We want to compare two different approaches for the computation of different contributions to the EM field:

- RF
- Beam
- ECloud

} EM solver

- RF
- Beam

} EM solver

- ECloud

} ES solver

To do so we developed self-consistent simulations, to make sure that we are not missing any physics.

Outline

Self-consistent Simulations

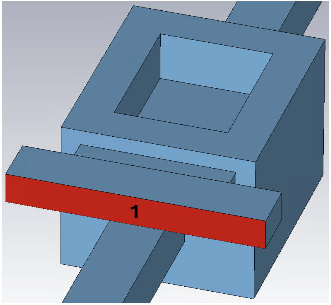
For a truly self-consistent simulation we need to excite the RF field directly in Warp.

Self-consistent Simulations

For a truly self-consistent simulation we need to excite the RF field directly in Warp.

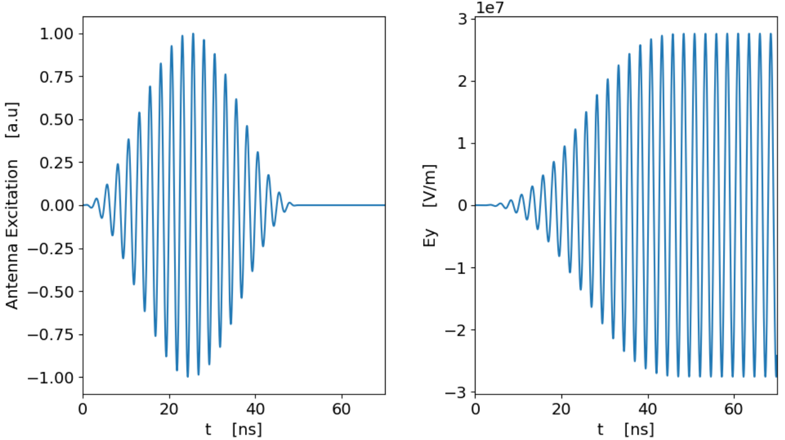
In Warp we can feed the cavity through a waveguide in which we place a "laser antenna" (which is normally used in laser-plasma simulations):

- It consists of a time-varying current sheet which excites the cavity mode.
- Geometrically it reminds of the feeder of the RFD



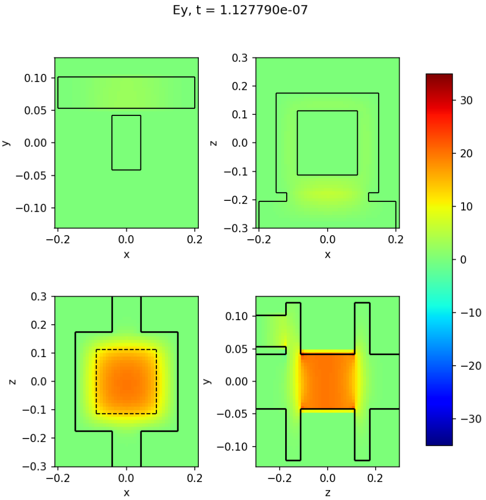
The Field in the Cavity

We probe E_y at the cavity center to visualize its time evolution compared to the antenna excitation.



The electric field increases when the antenna is on and keeps resonating as the antenna is turned off. As the cavity is lossless we can keep the antenna off for the rest of the simulation.

RF Mode of the Crab Cavity - Results



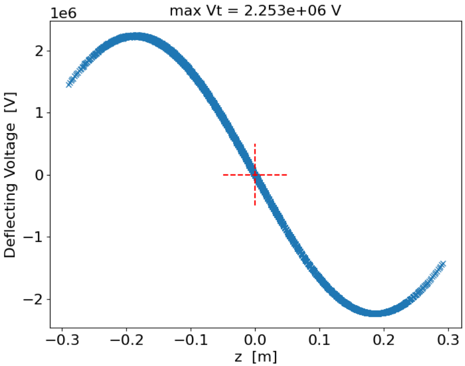
MOVIE

Bunch Deflection - Transverse Kick

A way to test our computation of the RF fields is to measure the deflecting voltage directly on a p^+ bunch.

$$V_t = \frac{E_{beam}}{q_e} \Delta y' \qquad E_{beam} = \text{beam energy}$$

This test is very useful to phase the bunches correctly with the cavity.

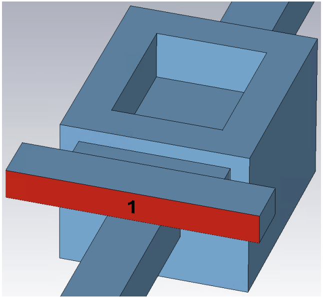


We clearly see that the head and tail of the bunches are kicked in opposite directions.

Outline of a Self-Consistent Ecloud Simulations

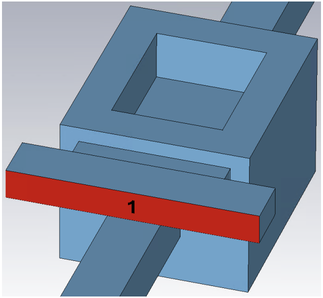
Outline of a Self-Consistent Ecloud Simulations

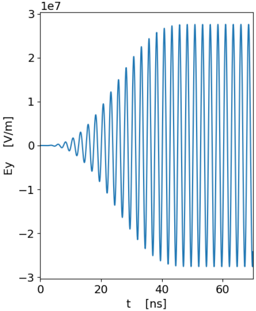
- Initialize the simulation



Outline of a Self-Consistent Ecloud Simulations

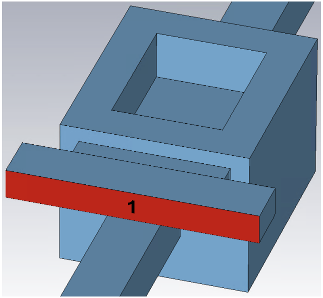
- Initialize the simulation
- Turn on the antenna and simulate the transient that excites the cavity mode

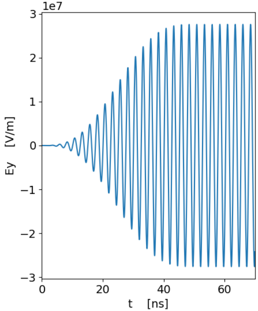




Outline of a Self-Consistent Ecloud Simulations

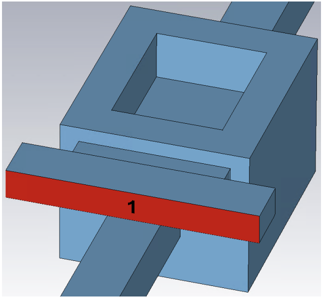
- Initialize the simulation
- Turn on the antenna and simulate the transient that excites the cavity mode
- Turn off the antenna

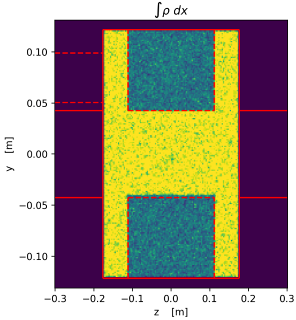


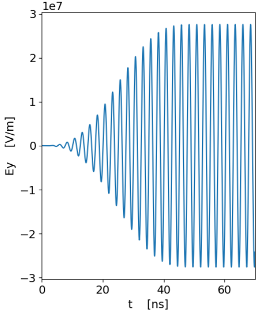


Outline of a Self-Consistent Ecloud Simulations

- Initialize the simulation
- Turn on the antenna and simulate the transient that excites the cavity mode
- Turn off the antenna
- Introduce s uniform distribution of seed electrons

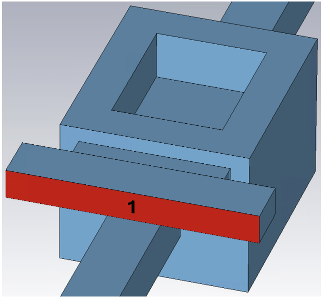


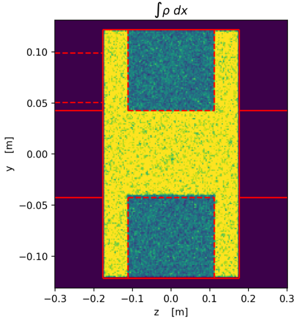


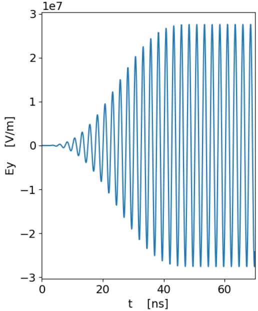


Outline of a Self-Consistent Ecloud Simulations

- Initialize the simulation
- Turn on the antenna and simulate the transient that excites the cavity mode
- Turn off the antenna
- Introduce s uniform distribution of seed electrons
- Inject bunches



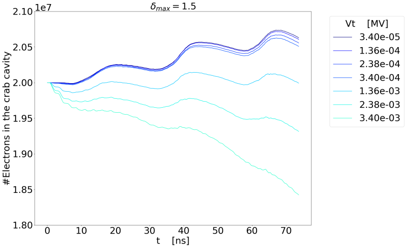




Outline

Build-Up VS Transverse Voltage

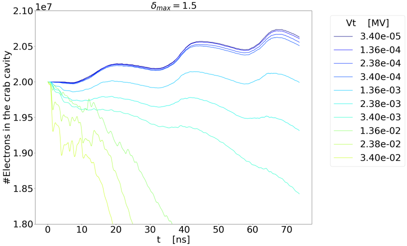
We simulate the ECloud buildup for increasing transverse voltages.



- Low voltages: beam-induced multipacting is observed

Build-Up VS Transverse Voltage

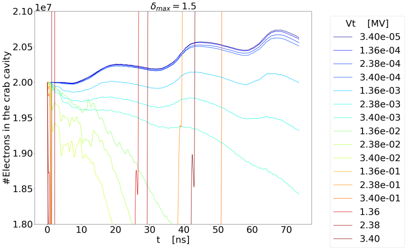
We simulate the ECloud buildup for increasing transverse voltages.



- Low voltages: beam-induced multipacting is observed
- Medium voltages: the RF field suppresses the beam-induced multipacting

Build-Up VS Transverse Voltage

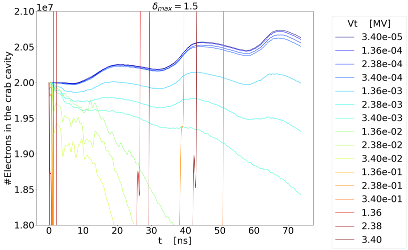
We simulate the ECloud buildup for increasing transverse voltages.

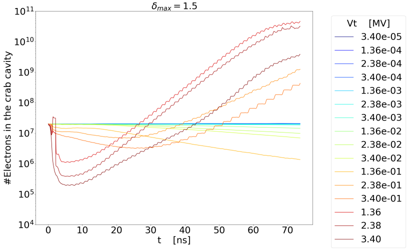


- Low voltages: beam-induced multipacting is observed
- Medium voltages: the RF field suppresses the beam-induced multipacting
- High voltages (up to nominal): RF-induced multipacting arises

Build-Up VS Transverse Voltage

We simulate the ECloud buildup for increasing transverse voltages.



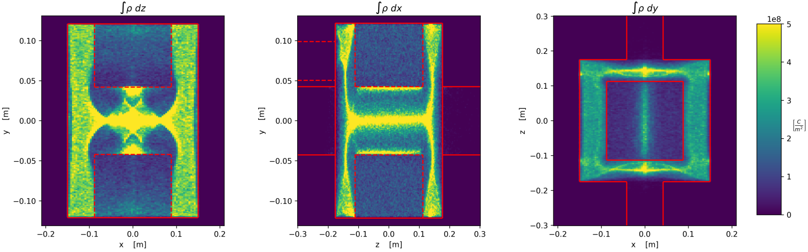


- Low voltages: beam-induced multipacting is observed
- Medium voltages: the RF field suppresses the beam-induced multipacting
- High voltages (up to nominal): RF-induced multipacting arises

We visualize the electrons distribution, in order to have a better understanding of the electrons dynamics.

Electrons Distribution - $V_t = 34MV$

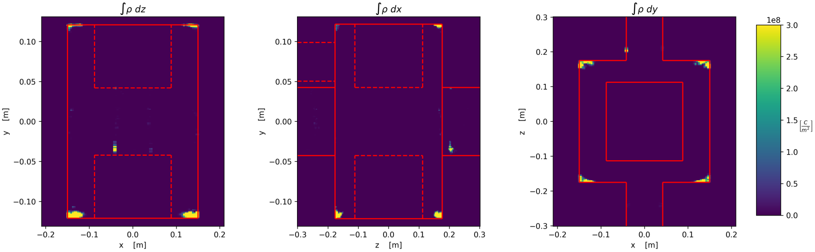
For $V_t = 0V$ we have beam-induced multipacting between the cavity poles.



MOVIE

Electrons Distribution - $V_t = 3.4MV$

We see that for $V_t = 3.4MV$ the electrons tend to form clusters in the corners.

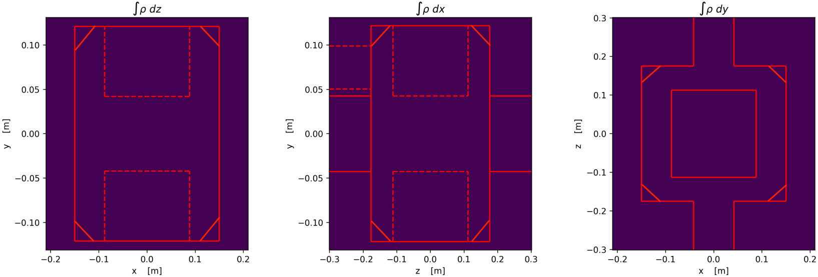


Given that the corners are not present in the actual cavities this result is not very interesting.

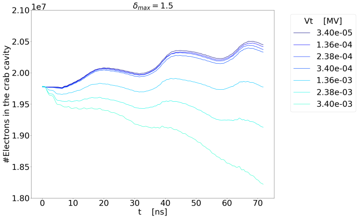
MOVIE

Carving the corners

To avoid clusters in the corners, we prevent the electrons from reaching these areas inserting additional planes into the domain.

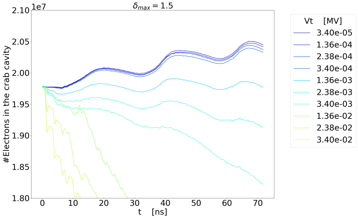


Carving the corners - Results



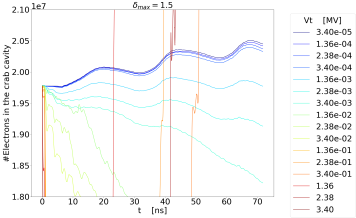
- Low voltages: beam-induced multipacting is observed

Carving the corners - Results



- Low voltages: beam-induced multipacting is observed
- Medium voltages: the RF field suppresses the beam-induced multipacting

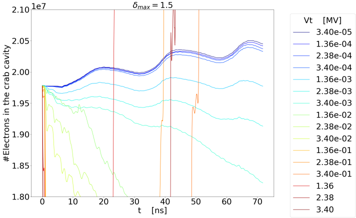
Carving the corners - Results

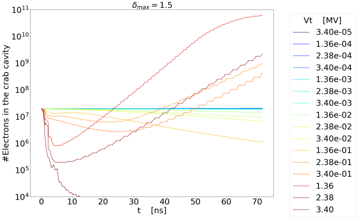


- Low voltages: beam-induced multipacting is observed
- Medium voltages: the RF field suppresses the beam-induced multipacting
- High voltages (up to nominal): RF-induced multipacting arises

Let's visualize again the distribution of the electrons.

Carving the corners - Results



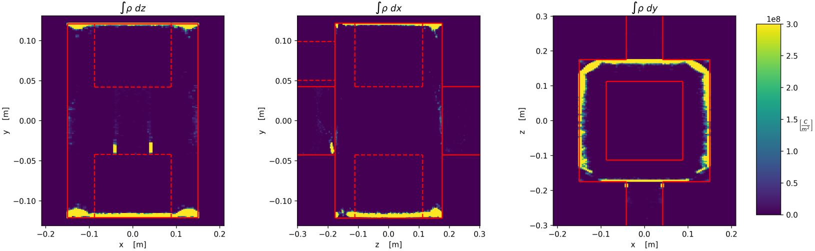


- Low voltages: beam-induced multipacting is observed
- Medium voltages: the RF field suppresses the beam-induced multipacting
- High voltages (up to nominal): RF-induced multipacting arises

Let's visualize again the distribution of the electrons.

Electrons Distribution

The electrons are concentrated around the upper/lower edges of the cavity.



This is compatible with what has been observed in [1]

[1] Andrés et al., "Design and vertical tests of double-quarter wave cavity prototypes for the high-luminosity LHC crab cavity system".

Electromagnetic or Electrostatic ECloud Field?

Remember that we want to compare the following approaches for the computation of the each contribution to the EM field:

- RF
- Beam
- ECloud

} EM solver

- RF
- Beam

} EM solver

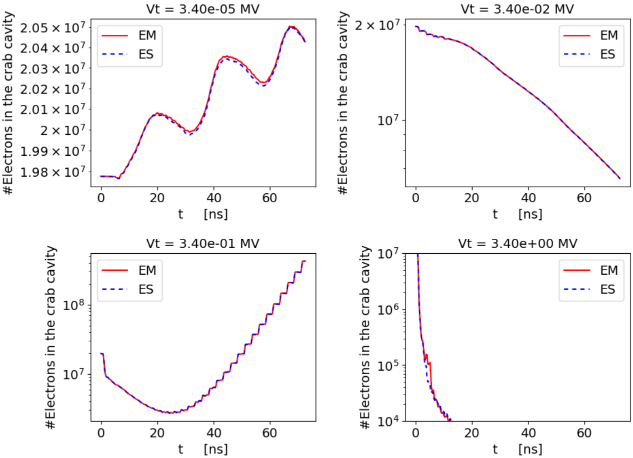
- ECloud

} ES solver

We implemented both of the approaches and in the following we compare them.

Electrostatic VS Electromagnetic

Plots for increased values of the deflecting voltage:



The ES and EM solvers agree really well, thus we conclude that the self-interaction of the electrons can be approximated as electrostatic.
This is particularly convenient in sight of computations in realistic geometries (as the EM solver in Warp doesn't handle properly curved boundaries).

Outline

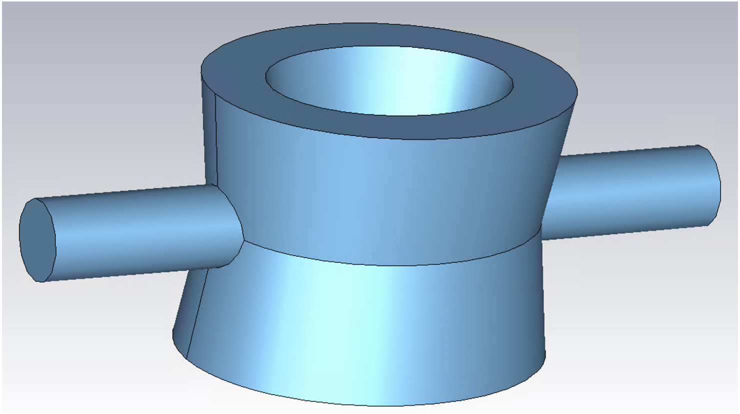
Realistic Geometry

We started working on a more realistic geometry, therefore we modelled a DQW-like cavity.

The main differences with the actual design are:

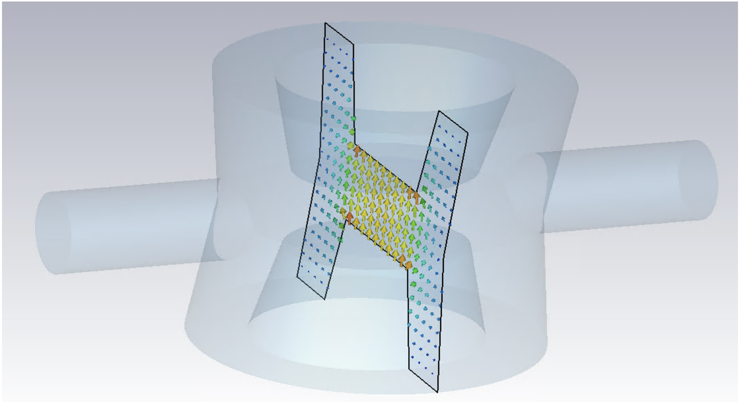
- Sharp edges (no weldings)
- Missing FPC, HOM couplers..

These simplifications are made to speed up the simulations, but the procedure we will follow can be extended to the full model.



Tuning the Realistic Cavity

We used the eigensolver to check the fundamental frequency of the cavity.

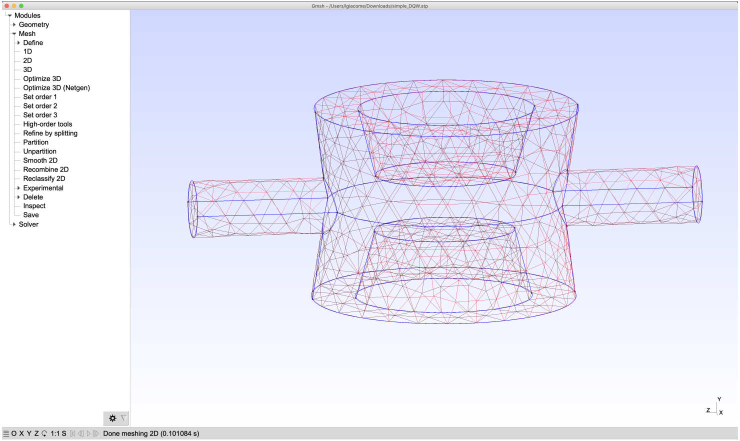


The height of the cavity has been slightly adjusted to achieve the correct fundamental frequency (400MHz).

From CST to Warp

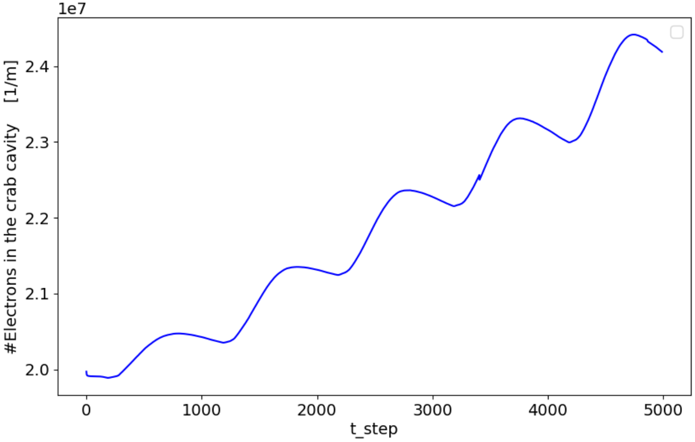
Workflow:

$$CST \xrightarrow{.stp} Gmsh \xrightarrow{.msh} MeshIO \longrightarrow Warp$$



ECloud Build-up

As a first step, we simulated the E-Cloud build-up in the realistic DQW cavity without RF fields.



The next step will be to insert the RF fields from CST.

Summary and Next Steps

Summary and Next Steps

- In collaboration with LBNL we developed a framework for ECloud simulations in very general configurations (Warp + PyECLOUD);
  - 3D geometries and externally applied RF fields;

Summary and Next Steps

- In collaboration with LBNL we developed a framework for ECloud simulations in very general configurations (Warp + PyECLOUD);
  - 3D geometries and externally applied RF fields;
- Investigated two approaches for the simulations in a simplified geometry;
  - Pre-computed unperturbed RF fields + quasi-static solver for the electrons self-fields;
  - Self-consistent simulations (with RF field computed in Warp);

Summary and Next Steps

- In collaboration with LBNL we developed a framework for ECloud simulations in very general configurations (Warp + PyECLOUD);
  - 3D geometries and externally applied RF fields;
- Investigated two approaches for the simulations in a simplified geometry;
  - Pre-computed unperturbed RF fields + quasi-static solver for the electrons self-fields;
  - Self-consistent simulations (with RF field computed in Warp);
- Tests have shown that the two approaches are equivalent for the cases of interest;

Summary and Next Steps

- In collaboration with LBNL we developed a framework for ECloud simulations in very general configurations (Warp + PyECLOUD);
  - 3D geometries and externally applied RF fields;
- Investigated two approaches for the simulations in a simplified geometry;
  - Pre-computed unperturbed RF fields + quasi-static solver for the electrons self-fields;
  - Self-consistent simulations (with RF field computed in Warp);
- Tests have shown that the two approaches are equivalent for the cases of interest;
- Simulated beam-induced multipacting in a realistic DQW cavity;

Summary and Next Steps

- In collaboration with LBNL we developed a framework for ECloud simulations in very general configurations (Warp + PyECLOUD);
  - 3D geometries and externally applied RF fields;
- Investigated two approaches for the simulations in a simplified geometry;
  - Pre-computed unperturbed RF fields + quasi-static solver for the electrons self-fields;
  - Self-consistent simulations (with RF field computed in Warp);
- Tests have shown that the two approaches are equivalent for the cases of interest;
- Simulated beam-induced multipacting in a realistic DQW cavity;
- Presently, we are working at the simulation of a more realistic geometry
  - Simulations for $V_{RF} = 0$ already performed;
  - to be extended to cover the full realistic case