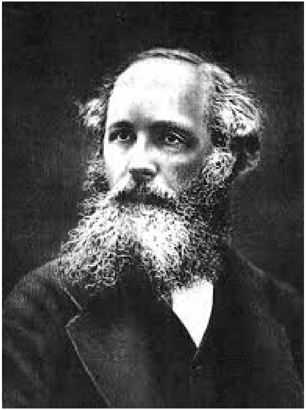
CERN

ETH zürich

Effect of E-Cloud Magnetic Field on Protons and Electrons Motion

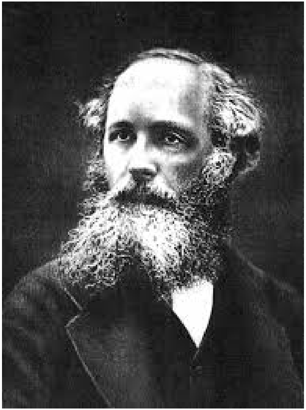
L. Giacomel, G. Iadarola

Many thanks to: K. Paraschou, L. Sabato



Siméon-Denis Poisson

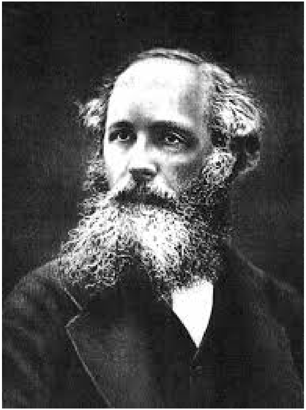
James Clerk Maxwell



Siméon-Denis Poisson

Electrostatics

James Clerk Maxwell



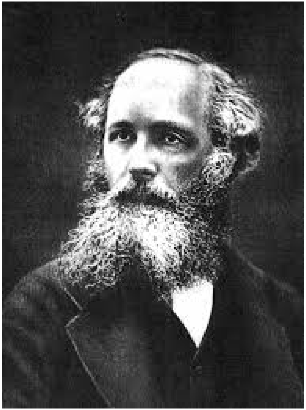
Electrodynamics

Siméon-Denis Poisson

Electrostatics

VS

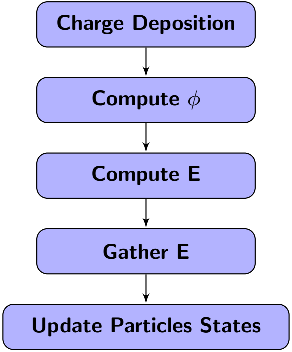
James Clerk Maxwell

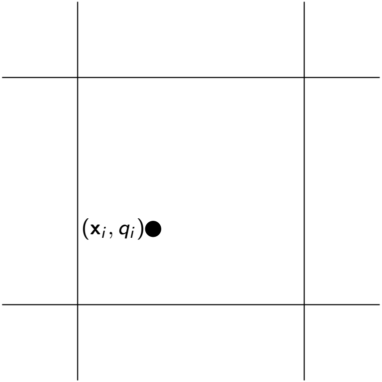


Electrodynamics

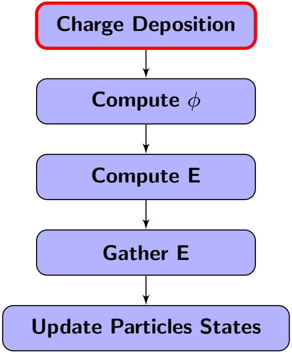
We will investigate which of the two models suits best our 2D e-cloud simulations.

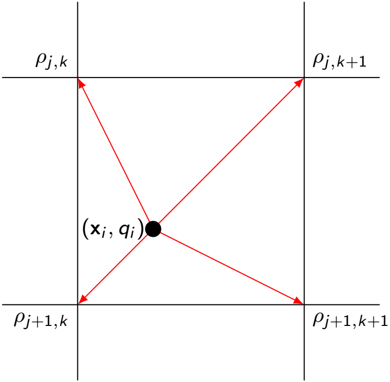
PyECLOUD is an Electrostatic Code



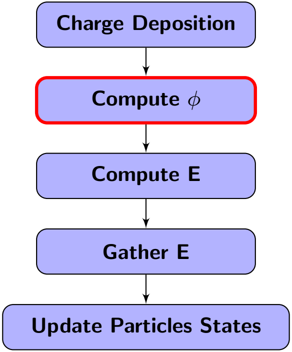


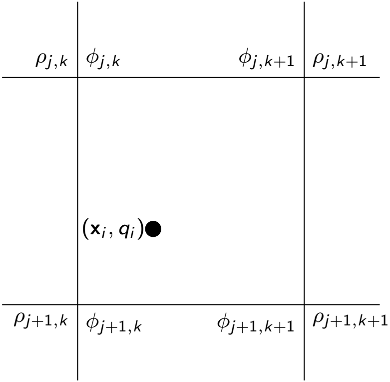
PyECLOUD is an Electrostatic Code





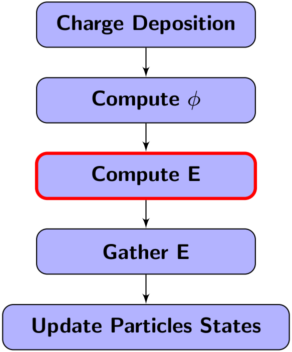
PyECLOUD is an Electrostatic Code

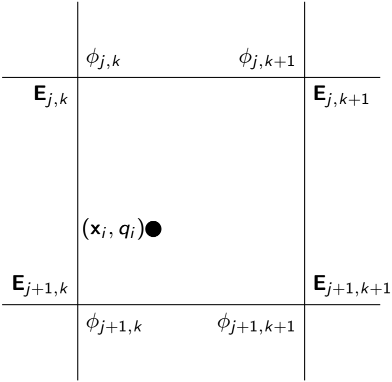




$$\nabla^2\phi = \rho$$

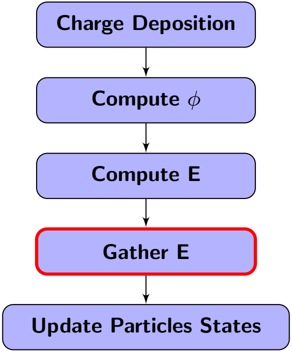
PyECLOUD is an Electrostatic Code

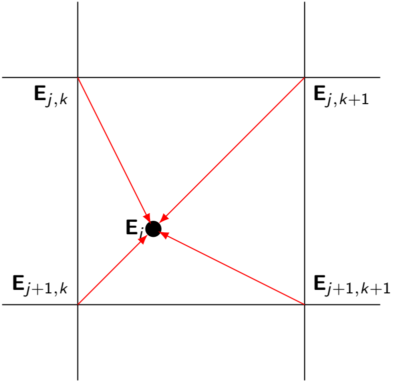




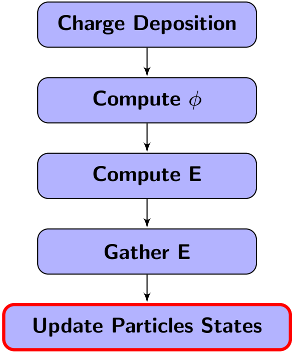
$$\mathbf{E} = -\nabla \phi$$

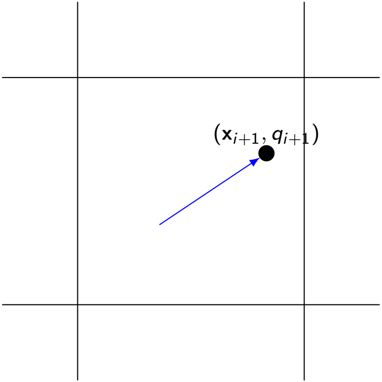
PyECLOUD is an Electrostatic Code





PyECLOUD is an Electrostatic Code





$$\begin{cases} \frac{d\mathbf{x}}{dt} = \mathbf{v} \\ \frac{d\mathbf{p}}{dt} = q(\mathbf{E} + \mathbf{v} \times \mathbf{B}^{EXT}) \end{cases}$$

Is Electrostatics Good Enough?

For e-cloud simulations in long structures with a static magnetic field (drifts, dipoles, quadrupoles...) 2D electrostatic simulations are commonly used.

Are we missing any physical behavior when using electrostatic simulations?

We would like to carry out electromagnetic simulations to compare with our electrostatic results.

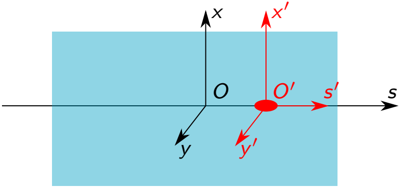
Is Electrostatics Good Enough?

For e-cloud simulations in long structures with a static magnetic field (drifts, dipoles, quadrupoles...) 2D electrostatic simulations are commonly used.

Are we missing any physical behavior when using electrostatic simulations?

We would like to carry out electromagnetic simulations to compare with our electrostatic results.

Useful tool: Electromagnetic Modelling in a Boosted Frame[1]



Main message: a bunch travelling (nearly) at the speed of light sees ρ' and $\mathbf{J}'$ of the e-cloud as stationary.

Therefore, $\mathbf{E}'$ and $\mathbf{B}'$ are decoupled and can be found by solving electrostatic and magnetostatic problems.

[1] G. Iadarola

Boosted Electro/Magneto-static

Since the sources are stationary the electric and magnetic field are decoupled and they are solution of the equations

$$\begin{cases} \nabla' \times \mathbf{E}' = \mathbf{0} \\ \nabla' \cdot \mathbf{E}' = \frac{\rho'}{\epsilon_0} \end{cases} \qquad \begin{cases} \nabla' \times \mathbf{B}' = \mu_0 \mathbf{J}' \\ \nabla' \cdot \mathbf{B}' = 0 \end{cases}$$

Boosted Electro/Magneto-static

Since the sources are stationary the electric and magnetic field are decoupled and they are solution of the equations

$$\begin{cases} \nabla' \times \mathbf{E}' = \mathbf{0} \\ \nabla' \cdot \mathbf{E}' = \frac{\rho'}{\epsilon_0} \end{cases} \qquad \begin{cases} \nabla' \times \mathbf{B}' = \mu_0 \mathbf{J}' \\ \nabla' \cdot \mathbf{B}' = 0 \end{cases}$$

If we express the fields in terms of potentials

$$\mathbf{E}' = -\nabla' \phi' \qquad \mathbf{B}' = \nabla' \times \mathbf{A}'$$

Boosted Electro/Magneto-static

Since the sources are stationary the electric and magnetic field are decoupled and they are solution of the equations

$$\begin{cases} \nabla' \times \mathbf{E}' = \mathbf{0} \\ \nabla' \cdot \mathbf{E}' = \frac{\rho'}{\epsilon_0} \end{cases} \qquad \Bigg| \qquad \begin{cases} \nabla' \times \mathbf{B}' = \mu_0 \mathbf{J}' \\ \nabla' \cdot \mathbf{B}' = 0 \end{cases}$$

If we express the fields in terms of potentials

$$\mathbf{E}' = -\nabla' \phi' \qquad \Big| \qquad \mathbf{B}' = \nabla' \times \mathbf{A}'$$

we easily find that the potentials solve the differential problems

$$\begin{cases} \nabla'^2 \phi' = -\frac{\rho'}{\epsilon_0} \\ + \text{ Dirichlet B.C.} \end{cases} \qquad \Bigg| \qquad \begin{cases} \nabla'^2 \mathbf{A}' = -\mu_0 \mathbf{J} \\ + \text{ Dirichlet B.C.} \end{cases}$$

Boosted Electro/Magneto-static

Since the sources are stationary the electric and magnetic field are decoupled and they are solution of the equations

$$\begin{cases} \nabla' \times \mathbf{E}' = \mathbf{0} \\ \nabla' \cdot \mathbf{E}' = \frac{\rho'}{\epsilon_0} \end{cases} \qquad \Bigg| \qquad \begin{cases} \nabla' \times \mathbf{B}' = \mu_0 \mathbf{J}' \\ \nabla' \cdot \mathbf{B}' = 0 \end{cases}$$

If we express the fields in terms of potentials

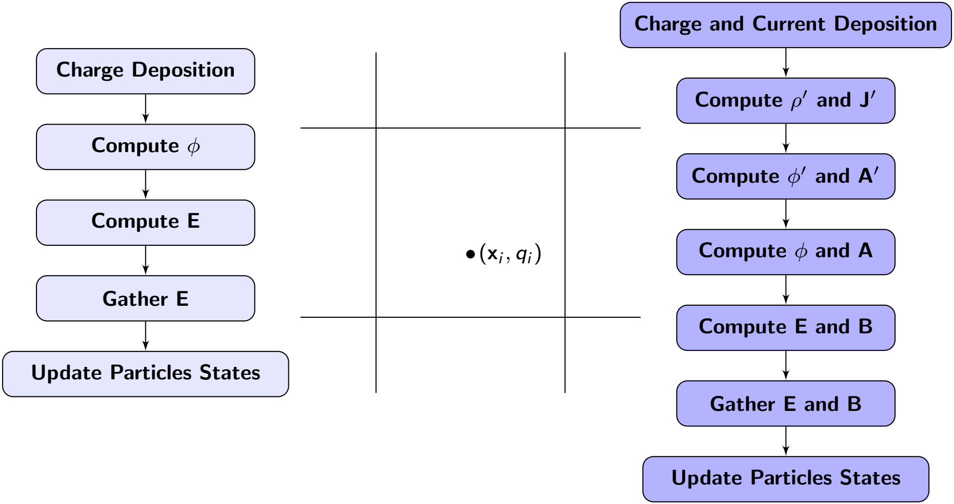
$$\mathbf{E}' = -\nabla' \phi' \qquad \Big| \qquad \mathbf{B}' = \nabla' \times \mathbf{A}'$$

we easily find that the potentials solve the differential problems

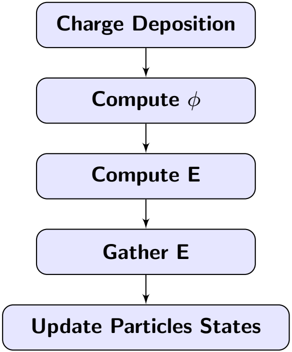
$$\begin{cases} \nabla'^2 \phi' = -\frac{\rho'}{\epsilon_0} \\ + \text{Dirichlet B.C.} \end{cases} \qquad \Bigg| \qquad \begin{cases} \nabla'^2 \mathbf{A}' = -\mu_0 \mathbf{J} \\ + \text{Dirichlet B.C.} \end{cases}$$

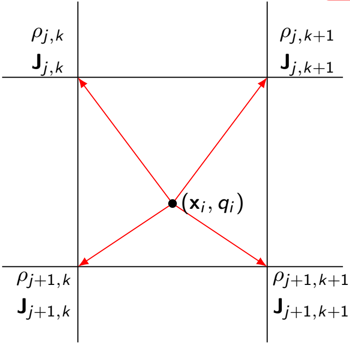
We already know how to solve these problems (i.e. use PyPIC).

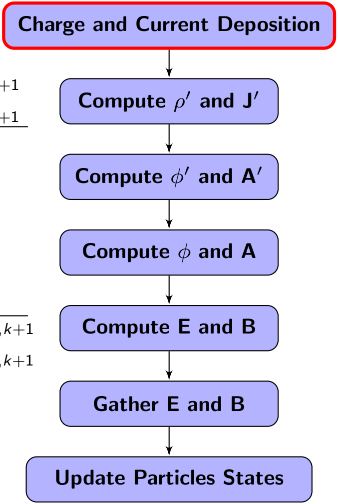
Boosted Electromagnetic PIC Iteration



Boosted Electromagnetic PIC Iteration

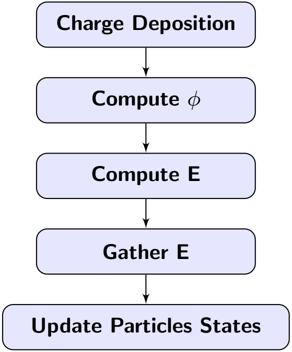


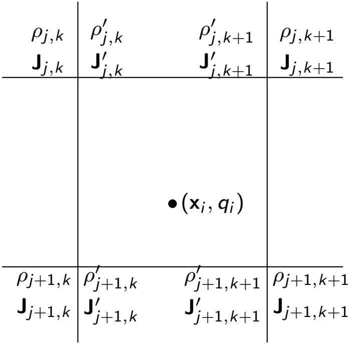


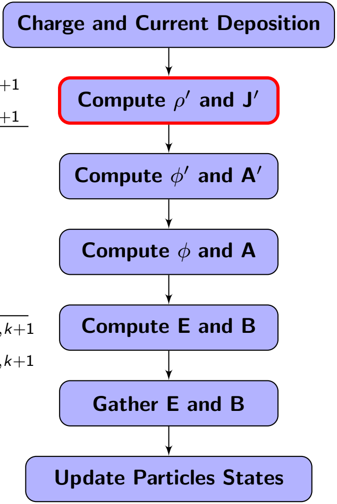


For a particle current computed as $q\mathbf{v}$

Boosted Electromagnetic PIC Iteration



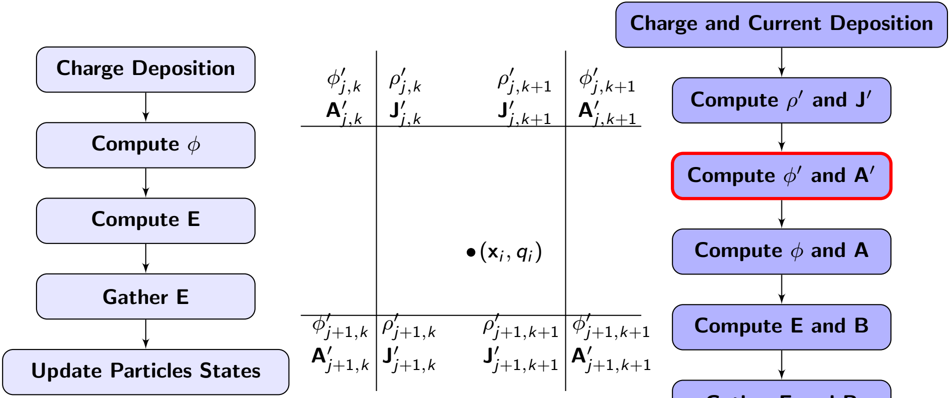




$$\rho' = \gamma\rho \qquad J'_s = \gamma\beta c\rho$$

$$J'_x = J_x \qquad J'_y = J_y$$

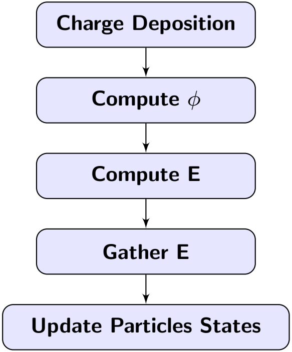
Boosted Electromagnetic PIC Iteration

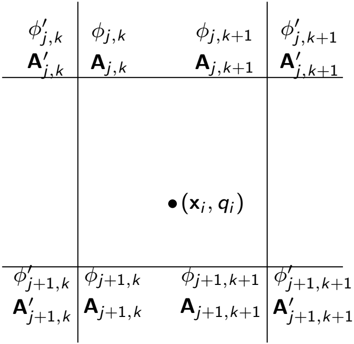


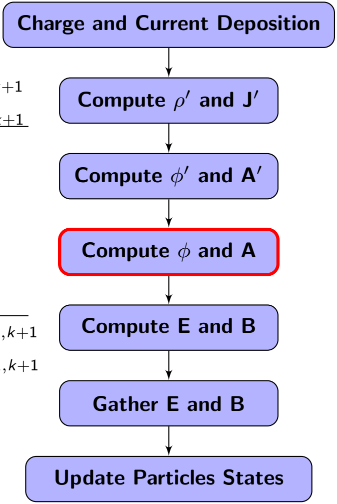
$$\nabla^2\phi' = -\frac{\rho'}{\epsilon_0}$$

$$\nabla^2\mathbf{A}' = -\mu_0\mathbf{J}'$$

Boosted Electromagnetic PIC Iteration



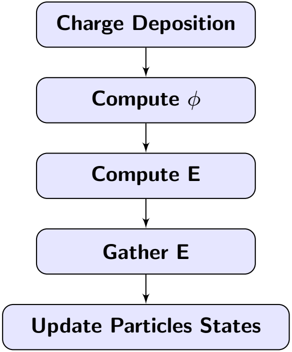


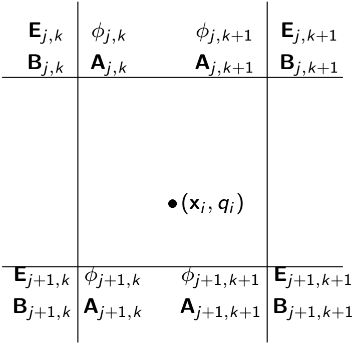


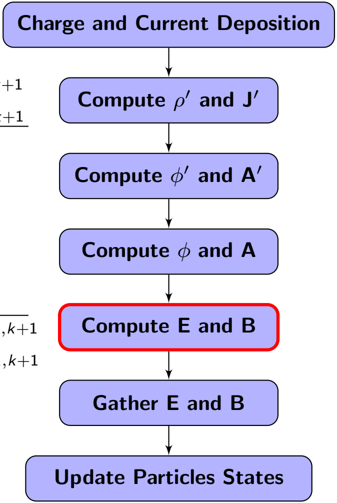
$$\phi = \frac{\phi'}{\gamma} \qquad A_x = A'_x$$

$$A_s = 0 \qquad A_y = A'_y$$

Boosted Electromagnetic PIC Iteration





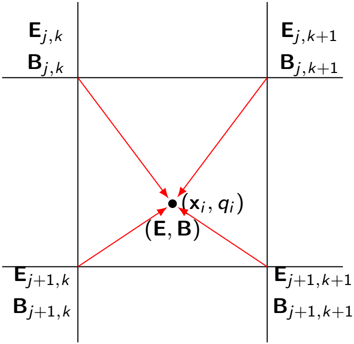


$$\mathbf{E} = -\nabla\phi - \frac{\partial \mathbf{A}}{\partial t}$$

$$\mathbf{B} = \nabla \times \mathbf{A}$$

Boosted Electromagnetic PIC Iteration

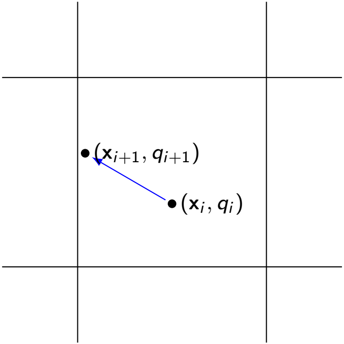
Charge Deposition
↓
Compute ϕ
↓
Compute **E**
↓
Gather **E**
↓
Update Particles States



Charge and Current Deposition
↓
Compute ρ' and $\mathbf{J}'$
↓
Compute ϕ' and $\mathbf{A}'$
↓
Compute ϕ and **A**
↓
Compute **E** and **B**
↓
Gather E and B
↓
Update Particles States

Boosted Electromagnetic PIC Iteration

Charge Deposition
↓
Compute ϕ
↓
Compute E
↓
Gather E
↓
Update Particles States



Charge and Current Deposition
↓
Compute ρ' and $\mathbf{J}'$
↓
Compute ϕ' and $\mathbf{A}'$
↓
Compute ϕ and $\mathbf{A}$
↓
Compute E and B
↓
Gather E and B
↓
Update Particles States

$$\begin{cases} \frac{d\mathbf{x}}{dt} = \mathbf{v} \\ \frac{d\mathbf{p}}{dt} = q(\mathbf{E} + \mathbf{v} \times (\mathbf{B} + \mathbf{B}^{EXT})) \end{cases}$$

Completed Boris Tracker

The Boris Tracker is used in PyECLOUD[2] to update the particle states as follows:

$$\frac{\mathbf{x}_{i+1} - \mathbf{x}_i}{\Delta t} = \mathbf{v}_{i+1}$$

$$\mathbf{v}_{i+1} = \mathbf{v}^+ + \frac{q}{m}\mathbf{E}\frac{\Delta t}{2}$$

$$\frac{\mathbf{v}^+ - \mathbf{v}^-}{\Delta t} = \frac{q}{2mc}(\mathbf{v}^+ + \mathbf{v}^-) \times \mathbf{B}^{EXT}$$

$$\mathbf{v}^- = \mathbf{v}_i + \frac{q}{m}\mathbf{E}\frac{\Delta t}{2}$$

$$\mathbf{B}^{EXT} = \begin{bmatrix} B_x^{EXT} \\ B_y^{EXT} \\ 0 \end{bmatrix}$$

The solver had been implemented enforcing the simplifications given by $B_s^{EXT} = 0$.

[2]G. Iadarola, G. Rumolo

Completed Boris Tracker

The Boris Tracker is used in PyECLOUD[2] to update the particle states as follows:

$$\frac{\mathbf{x}_{i+1} - \mathbf{x}_i}{\Delta t} = \mathbf{v}_{i+1}$$

$$\mathbf{v}_{i+1} = \mathbf{v}^+ + \frac{q}{m}\mathbf{E}\frac{\Delta t}{2}$$

$$\frac{\mathbf{v}^+ - \mathbf{v}^-}{\Delta t} = \frac{q}{2mc}(\mathbf{v}^+ + \mathbf{v}^-) \times (\mathbf{B}^{EXT} + \mathbf{B})$$

$$\mathbf{v}^- = \mathbf{v}_i + \frac{q}{m}\mathbf{E}\frac{\Delta t}{2}$$

$$\mathbf{B}^{EXT} = \begin{bmatrix} B_x^{EXT} \\ B_y^{EXT} \\ 0 \end{bmatrix}$$

$$\mathbf{B} = \begin{bmatrix} B_x \\ B_y \\ B_s \end{bmatrix}$$

The solver had been implemented enforcing the simplifications given by $B_s^{EXT} = 0$.

[2] G. Iadarola, G. Rumolo

Completed Boris Tracker

The Boris Tracker is used in PyECLOUD[2] to update the particle states as follows:

$$\frac{\mathbf{x}_{i+1} - \mathbf{x}_i}{\Delta t} = \mathbf{v}_{i+1}$$

$$\mathbf{v}_{i+1} = \mathbf{v}^+ + \frac{q}{m}\mathbf{E}\frac{\Delta t}{2}$$

$$\frac{\mathbf{v}^+ - \mathbf{v}^-}{\Delta t} = \frac{q}{2mc}(\mathbf{v}^+ + \mathbf{v}^-) \times (\mathbf{B}^{EXT} + \mathbf{B})$$

$$\mathbf{v}^- = \mathbf{v}_i + \frac{q}{m}\mathbf{E}\frac{\Delta t}{2}$$

$$\mathbf{B}^{EXT} = \begin{bmatrix} B_x^{EXT} \\ B_y^{EXT} \\ 0 \end{bmatrix}$$

$$\mathbf{B} = \begin{bmatrix} B_x \\ B_y \\ B_s \end{bmatrix}$$

The solver had been implemented enforcing the simplifications given by $B_s^{EXT} = 0$. The code has now been modified to handle also the longitudinal component of the magnetic field.

[2] G. Iadarola, G. Rumolo

Boosted Frame Approach in PyECLOUD

The ideas so far discussed have been implemented and tested in PyECLOUD. Electromagnetic simulations can be enabled by setting

flag_em_tracking = True

in the simulation_parameters.input file.
Remark: the tracking method must be BorisMultipole!

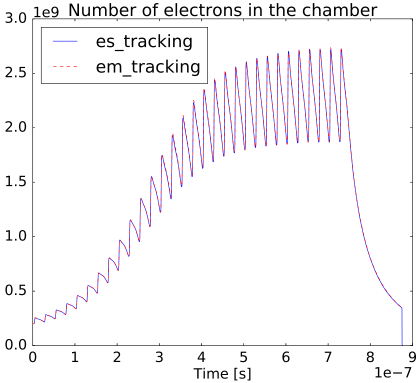
The implementation does not rely on any specific Poisson solver and can work with any solver that complies with the PyPIC interface.

Comparison With the ES Solver

We tested the implementation in the case of an LHC dipole magnet at injection energy ($\gamma = 479$) by comparing the results with the standard electrostatic solver.

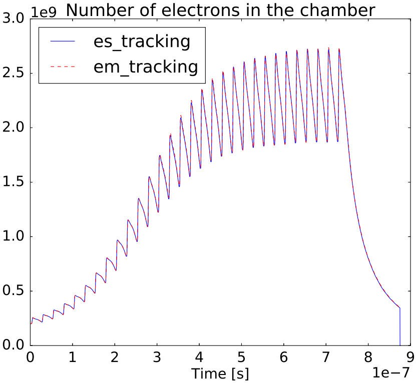
Comparison With the ES Solver

We tested the implementation in the case of an LHC dipole magnet at injection energy ($\gamma = 479$) by comparing the results with the standard electrostatic solver.



Comparison With the ES Solver

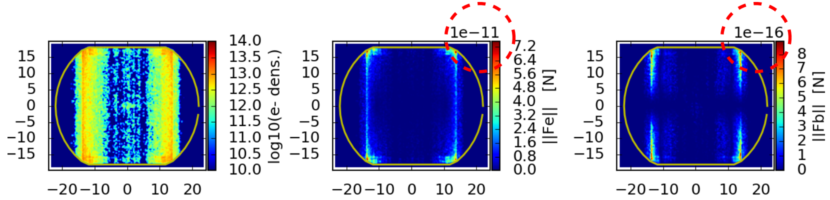
We tested the implementation in the case of an LHC dipole magnet at injection energy ($\gamma = 479$) by comparing the results with the standard electrostatic solver.



The results are extremely similar. It seems reasonable that in a dipole the e-cloud magnetic field is much less strong than the electric field.

F_e vs F_b

How relevant is the magnetic interaction?



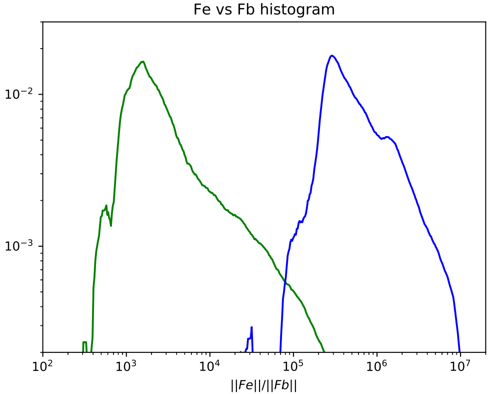
ANIMATION

F_e vs F_b

We can verify that in a dipole magnet the electrostatic solver does a good enough job because the electric interaction is much stronger than the magnetic one.

F_e vs F_b

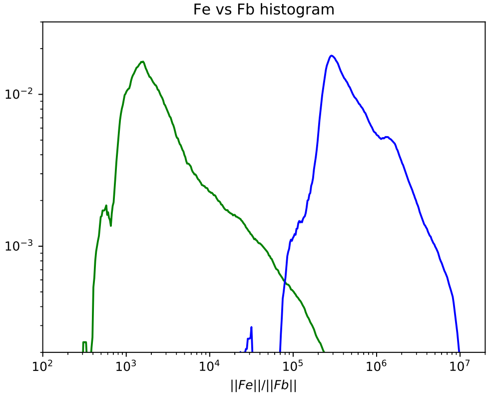
We can verify that in a dipole magnet the electrostatic solver does a good enough job because the electric interaction is much stronger than the magnetic one.



During the pinch, when the electrons are faster.

F_e vs F_b

We can verify that in a dipole magnet the electrostatic solver does a good enough job because the electric interaction is much stronger than the magnetic one.



During the pinch, when the electrons are faster.

Force Acting on the Beam

It has been proved that the kick on the beam generated by the e-cloud is only effect of ϕ.

Force Acting on the Beam

It has been proved that the kick on the beam generated by the e-cloud is only effect of ϕ.

Force acting on the beam:

$$F = q\left(-\nabla\phi - \frac{\partial \mathbf{A}}{\partial t} + \beta c \hat{\mathbf{i}}_s \times \mathbf{B}\right).$$

Force Acting on the Beam

It has been proved that the kick on the beam generated by the e-cloud is only effect of ϕ.

Force acting on the beam:

$$F = q\left(-\nabla\phi - \frac{\partial \mathbf{A}}{\partial t} + \beta c \hat{\mathbf{i}}_s \times \mathbf{B}\right).$$

It is possible to show analytically that

$$\frac{\partial \mathbf{A}}{\partial t} = \beta c \hat{\mathbf{i}}_s \times \mathbf{B}$$

and therefore

$$\mathbf{F} = -q\nabla\phi$$

We used this fact to cross check our implementation.

Force Acting on the Beam

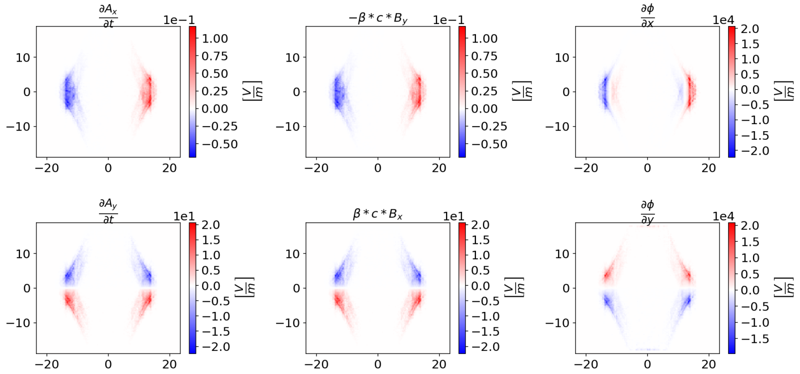
We verified numerically the relation

$$\frac{\partial \mathbf{A}}{\partial t} = \beta c \hat{\mathbf{i}}_s \times \mathbf{B} = \beta c \begin{bmatrix} -B_y \\ B_x \end{bmatrix}.$$

Force Acting on the Beam

We verified numerically the relation

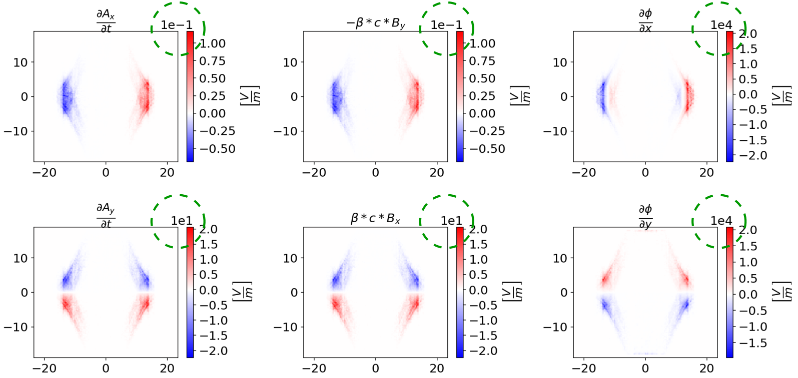
$$\frac{\partial \mathbf{A}}{\partial t} = \beta c \hat{\mathbf{i}}_s \times \mathbf{B} = \beta c \begin{bmatrix} -B_y \\ B_x \end{bmatrix}.$$



Force Acting on the Beam

We verified numerically the relation

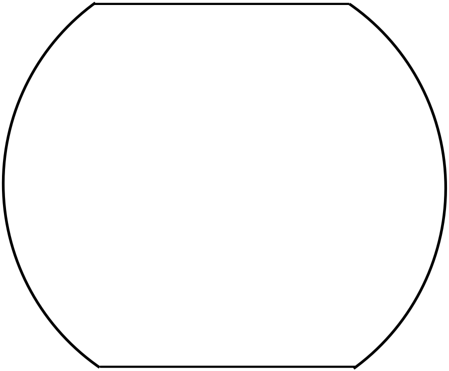
$$\frac{\partial \mathbf{A}}{\partial t} = \beta c \hat{\mathbf{i}}_s \times \mathbf{B} = \beta c \begin{bmatrix} -B_y \\ B_x \end{bmatrix}.$$



Moreover, we clearly see that the terms related to ϕ dominate the others.

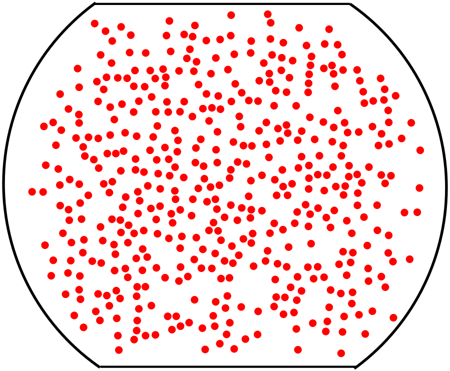
Force Acting on the Beam

We want to analyze more specifically the magnitude of the different components of the force acting on the beam in the beam area. We therefore probe them in a region with comparable size to the beam.



Force Acting on the Beam

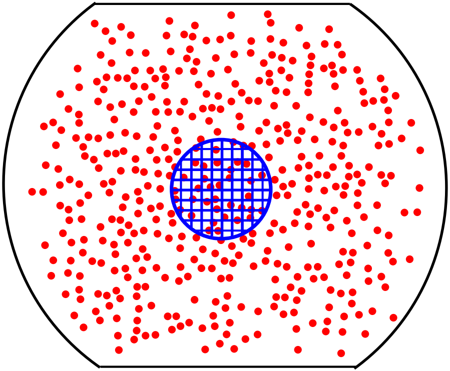
We want to analyze more specifically the magnitude of the different components of the force acting on the beam in the beam area. We therefore probe them in a region with comparable size to the beam.



Electron cloud particles

Force Acting on the Beam

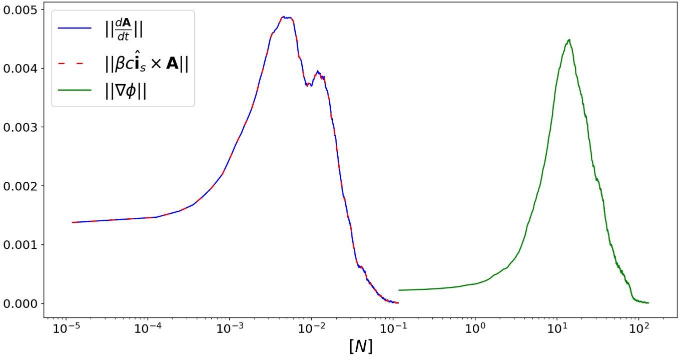
We want to analyze more specifically the magnitude of the different components of the force acting on the beam in the beam area. We therefore probe them in a region with comparable size to the beam.



Electron cloud particles
Uniform grid in an ellipse with axes $3*\sigma_x$ and $3*\sigma_y$

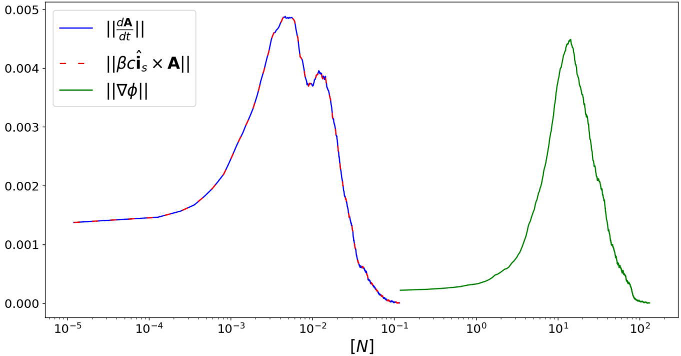
Force Acting on the Beam

We then measure the three components of the force caused by the e-cloud in the beam region.



Force Acting on the Beam

We then measure the three components of the force caused by the e-cloud in the beam region.



We observe a perfect overlap of $||\frac{\partial \mathbf{A}}{\partial t}||$ and $||\beta c\hat{\mathbf{i}}_s \times \mathbf{A}||$, while $||\nabla\phi||$ is 3 orders of magnitude bigger than the other two.

Conclusion

Conclusion

- We have added the possibility to carry out electromagnetic simulations in PyECLOUD

Conclusion

- We have added the possibility to carry out electromagnetic simulations in PyECLOUD
- Electromagnetic simulations are computationally more heavy

Conclusion

- We have added the possibility to carry out electromagnetic simulations in PyECLOUD
- Electromagnetic simulations are computationally more heavy
- The results of electrostatic and electromagnetic simulations are indistinguishable

Conclusion

- We have added the possibility to carry out electromagnetic simulations in PyECLOUD
- Electromagnetic simulations are computationally more heavy
- The results of electrostatic and electromagnetic simulations are indistinguishable
- Electrostatic simulations are good enough

Conclusion

- We have added the possibility to carry out electromagnetic simulations in PyECLOUD
- Electromagnetic simulations are computationally more heavy
- The results of electrostatic and electromagnetic simulations are indistinguishable
- Electrostatic simulations are good enough

home.cern