Space Charge Modules for PyHEADTAIL

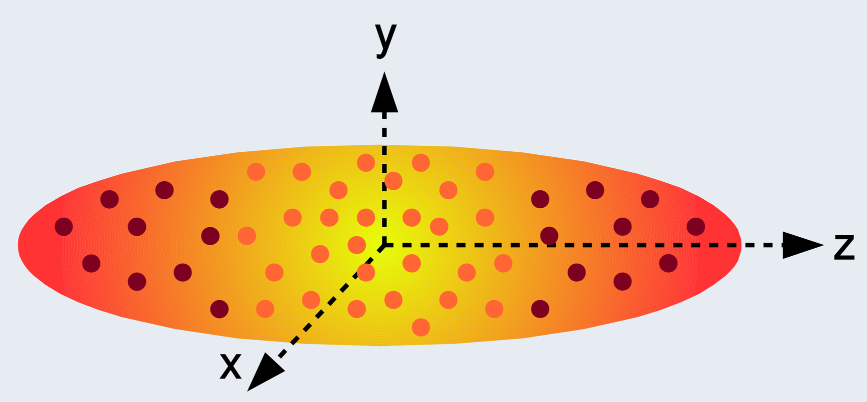
CPU and GPU Architectures, Incl. Particle-in-cell (2.5D and Full 3D)

A. Oeftiger[1], CERN, Switzerland; S. Hegglin, ETH Zürich, Switzerland

Abstract

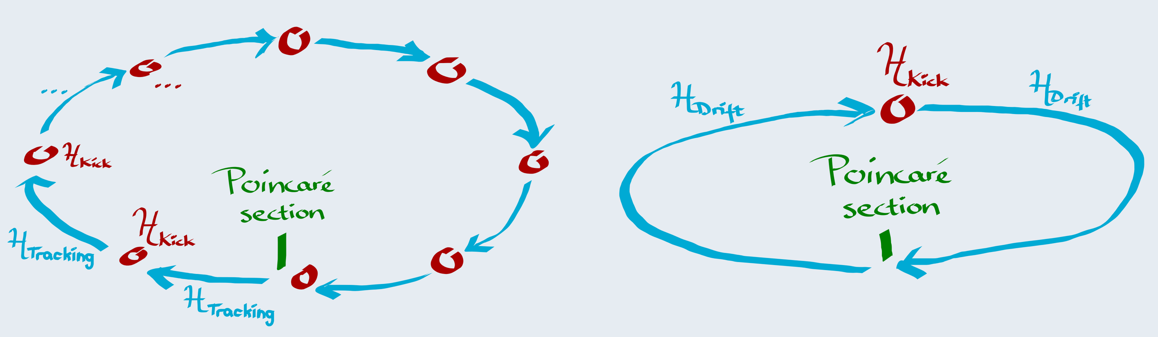
PyHEADTAIL is a 6D tracking tool developed at CERN to simulate collective effects. We present recent developments of the direct space charge (SC) suite, which is available for both the CPU and GPU. A new 3D particle-in-cell solver with open boundary conditions has been implemented. For the transverse plane, there is a semi-analytical Bassetti-Erskine model as well as 2D self-consistent particle-in-cell solvers with both open and closed boundary conditions. For the longitudinal plane, PyHEADTAIL offers line density derivative models. Simulations with these models are benchmarked with experiments at the injection plateau of CERN's SPS.

PyHEADTAIL



A PyHEADTAIL macro-particle beam of intensity N, particle charge q and particle mass m_p is described by the 6D set of coordinates $(x, x', y, y', z, \delta)$. *Single*-particle dynamics ("tracking") and *multi*-particle dynamics ("kicking") are separately solved in turns,

$$\mathcal{M}_{\text{rev}} = \exp\left(\Delta s :\mathcal{H}_{\text{tracking}}:\right) \exp\left(\Delta s :\mathcal{H}_{\text{kick}}:\right) \dots \quad (1)$$



(a) Transverse tracking. (b) Longitudinal tracking.

Figure 1 : PyHEADTAIL tracking model.

Transverse tracking based on linear Hill's equation: TWISS parameter based tracking provides correct beam sizes at kick points due to betatron $\beta_{x,y}(s)$ and dispersion $D_{x,y}(s)$ functions.

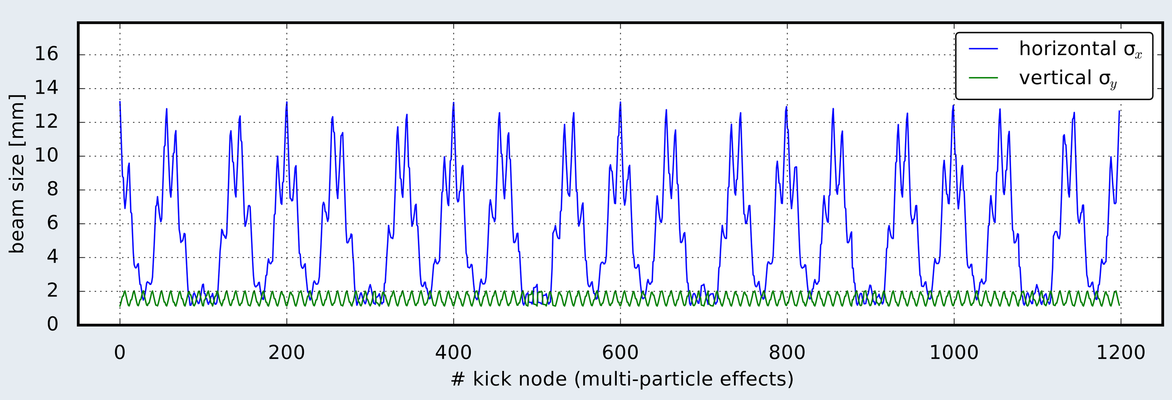


Figure 2 : High-brightness beam sizes for SPS Q20 optics.

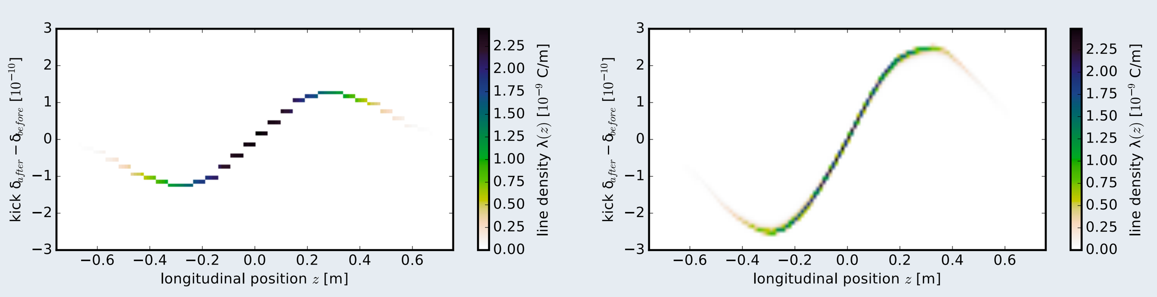
Particle-dependent phase advance $\psi_{x,y}(x, y, \delta; s)$ implements detuning from nth-order chromaticity and octupole fields. Longitudinal tracking has linear and sinusoidal multi-harmonic models. Multi-particle dynamics at kick points cover electron clouds, multi-polar wakefields (impedances) and *space charge*.

Longitudinal Space Charge

Line density derivative $\partial_z\lambda(z)$ model based on equivalent longitudinal electric field,

$$E_z^{\text{equiv}}(z) = -\frac{g}{4\pi\epsilon_0\gamma^2}\frac{d\lambda(z)}{dz} \quad . \quad (2)$$

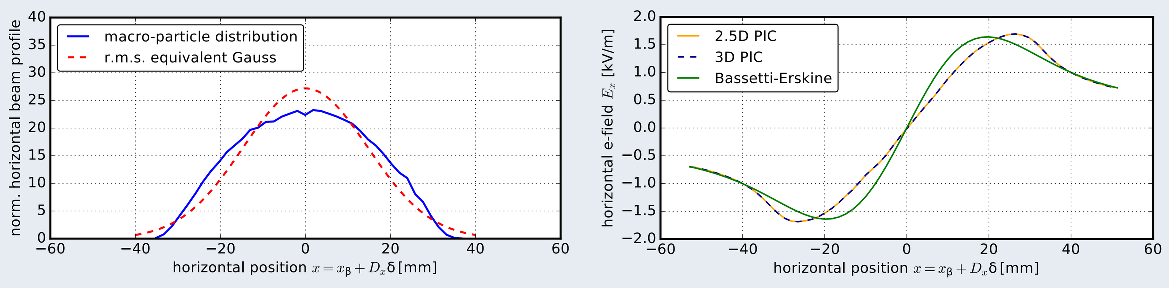
Averaged geometry factor g includes indirect SC, non-linear wall effects suppress $E_z(z)$ for long σ_z.



(a) $\partial_z\lambda(z)$ model. (b) (free-space) 3D PIC model.

Transverse Gaussian Space Charge

Bassetti-Erskine formula (cf. our paper) as 2.5D space charge model, applied slice-by-slice to the bunch distribution. Attention with dispersion and large longitudinal emittances ϵ_z: non-Gaussian δ distributions entail non-Gaussian horizontal beam profiles (despite a Gaussian betatron distribution)!



(a) Non-Gaussian hor. profile. (b) B.E. is approximative!

Self-consistent PIC Space Charge

Particle-in-cell (PIC) algorithms discretise the beam distribution onto a mesh:

1. particle to (regular) mesh deposition (P2M),
2. solve discrete Poisson equation

$$\Delta\phi = -\rho/\epsilon_0 \quad (3)$$

in *beam rest frame*, and

3. mesh to particle interpolation (M2P).

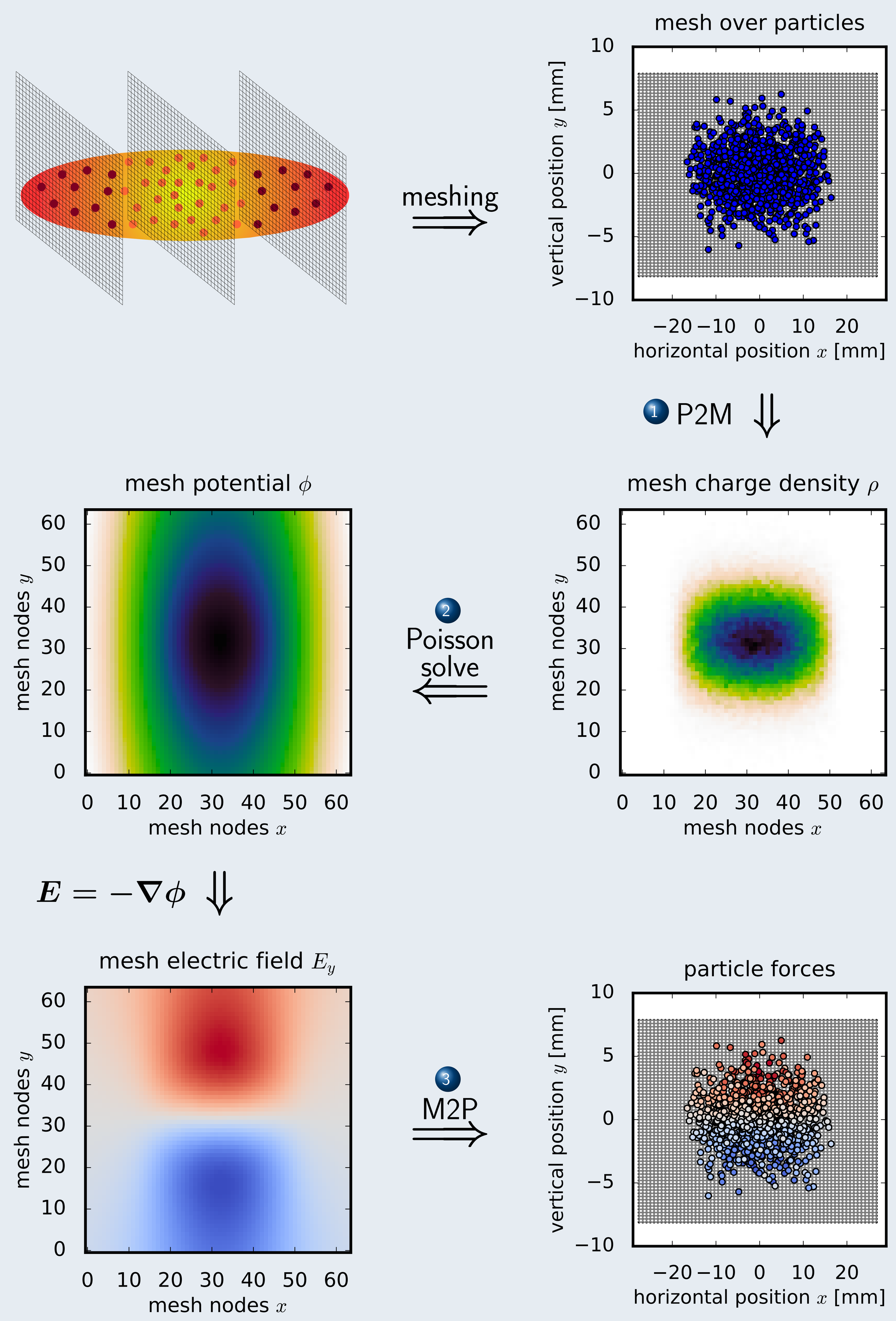


Figure 5 : Particle-in-cell algorithm for a 2.5D slice example.

Why Poisson Equation and not Full Maxwell?

In synchrotrons, relative momenta among particles are usually negligible. Hence, Lorentz boost from lab frame to beam rest frame and solve electrostatic problem $\Longrightarrow$ Poisson equation. Lorentz boost back to lab frame yields full electromagnetic fields (incl. transverse SC suppression from magnetic field). We implemented Poisson solvers for both 2.5D (slice-by-slice solving of transverse distributions) and 3D.

PIC: Finite Difference Poisson Solver

Direct matrix solving for nearest-neighbour sparse Poisson matrix (2D: 5-stencil, 3D: 7-stencil)

$$\mathcal{A}_{ij}\phi_j = -\rho_i/\epsilon_0 \quad . \quad (4)$$

- QR decomposition $\mathcal{A} = QR$ (orthogonal Q and upper R matrix, numerically extremely stable)
- LU decomposition $\mathcal{A} = LU$ (upper and lower triangle matrices L and U, faster than QR)

$\Longrightarrow$ finite difference requires boundary conditions!

PIC: Green's Function Poisson Solver

Use $D = 2$ or $D = 3$ Green's function $G(\mathbf{x}, \mathbf{y})$ from

$$\Delta G(\mathbf{x}) = \delta(\mathbf{x}) \quad (5)$$

to solve discrete Poisson equation for mesh potential

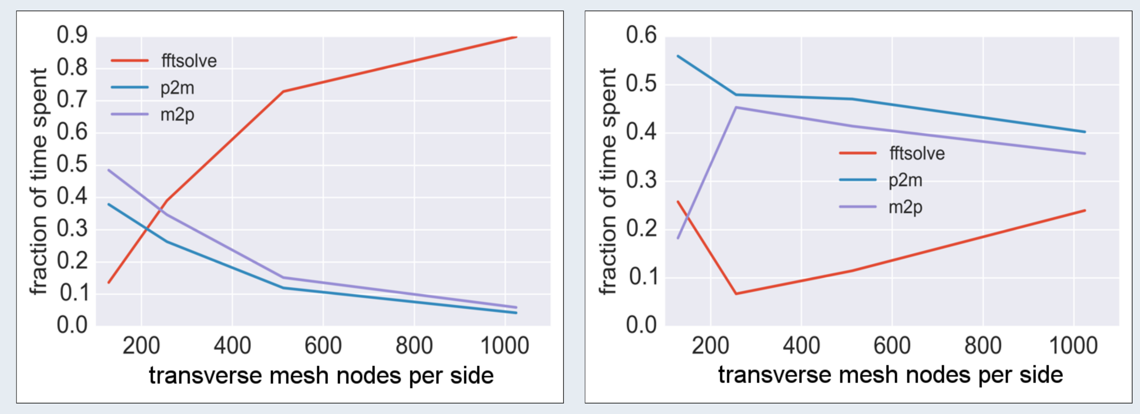
$$\phi(\mathbf{x}) = \frac{1}{2^{D-1}\pi\epsilon_0}\int d^D\hat{\mathbf{x}} \quad G(\hat{\mathbf{x}} - \mathbf{x})\,\rho(\hat{\mathbf{x}}) \quad . \quad (6)$$

(Hockney's) cyclic domain expansion allows rapid convolution via FFT algorithm. Also use "Integrated Green's Function" concept for large aspect ratios.

$\Longrightarrow$ free-space or rectangular boundary conditions!

GPU Acceleration

PyHEADTAIL and SC module have been parallelised for NVIDIA GPUs, $\Rightarrow$ PIC algorithm acceleration!



(a) CPU profile. (b) GPU profile.

Figure 6 : Different bottlenecks for runtime profiles of PIC.

On CPU, solve is bottleneck: FFT takes longer than mesh and particle interpolations. $\Rightarrow$ 3D space charge becomes prohibitive for long-term simulations!
On GPU, cuFFT accelerates by up to $S = 36$ for relevant meshes. 3D SC becomes feasible! In contrast, particle-to-mesh deposition becomes critical: $\mathcal{O}(1 \times 10^6)$ macro-particles each update 4x (2.5D) or 8x (3D) the $\mathcal{O}(1 \times 10^5)$ mesh memory entries.
We solved the corresponding thread stalls and memory bandwidth issues via pre-sorting of particle arrays by mesh node IDs (using `thrust::lower_bound` and `thrust::sort_by_key`).

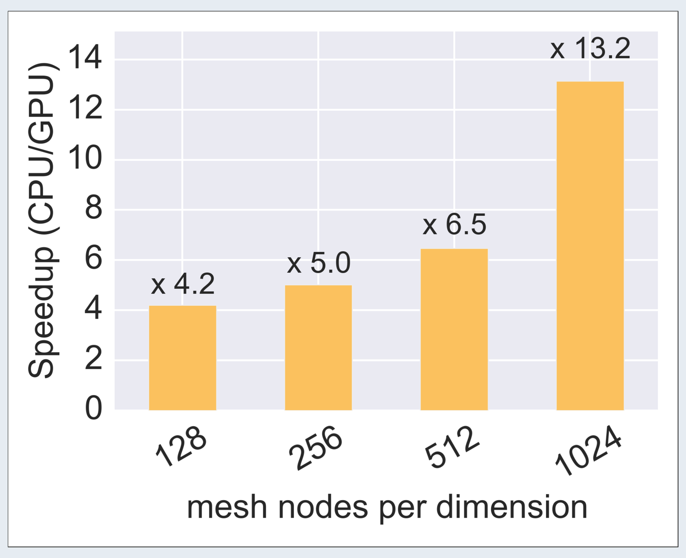


Figure 7 : Overall 2.5D PIC speed-up achieved vs. number of mesh nodes per transverse side comparing a NVIDIA K40m GPU to a single 2.3GHz Intel Xeon E5-2630 (v1) CPU core.

SPS Benchmark

High-brightness beams in SPS show influence of the octupolar $4Q_x = 81$ resonance. We drive the resonance with a single strong octupole to measure pronounced beam blow-up and losses $\Rightarrow$ benchmark for space charge simulations. Our beam has an incoherent transverse tune spread of $(-0.09, -0.16)$. The resonance causes a shifted blow-up peak at $Q_x = 20.28$, which is reproduced by simulations incl. TWISS lattice and Bassetti-Erskine SC.

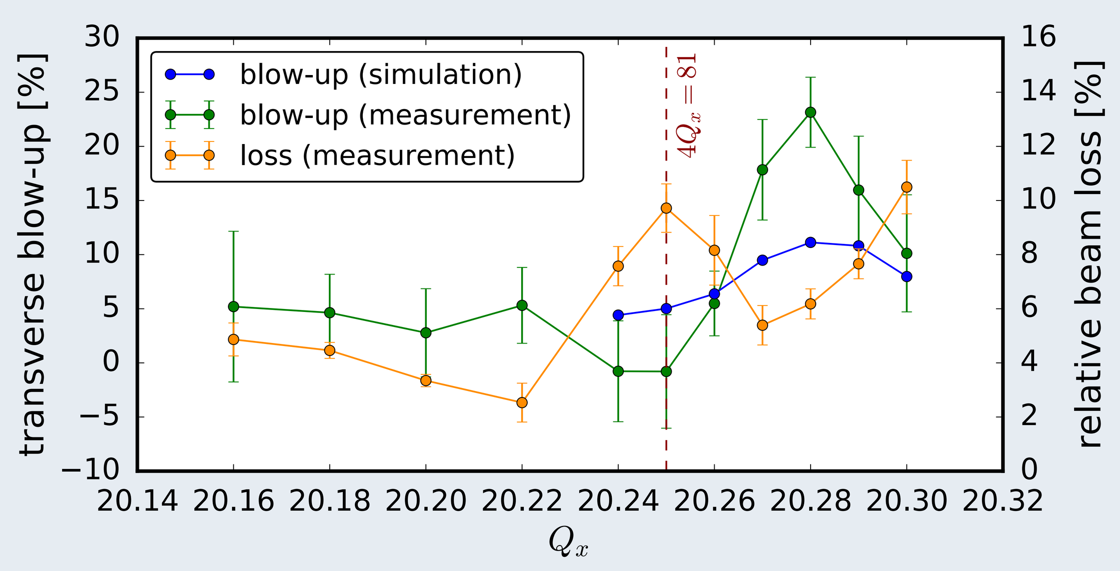


Figure 8 : Transverse emittance growth vs. coherent horizontal tune for Bassetti-Erskine space charge simulations over 10×10^3 turns and measurements over 130×10^3 turns.

HB'16 Contribution MOPR025, Malmö, Sweden, July 4, 2016

[1]adrian.oeftiger@cern.ch, also at LPAP, École Polytechnique Fédérale de Lausanne, Switzerland