Fast Self-Consistent Electron-Cloud Simulation

for CEPC in the EDR

Siyuan Feng/Yudong Liu

CEPC Workshop · Electron Cloud Session
October 2026

Outline

I. Method

- Where this code fits
- Why a boundary-only solver
- BEM: boundary integral $\rightarrow$ dense matrix
- FMM: tree and the six operators
- Coupling to the tracking loop

II. Results

- Validation vs. analytic image
- CEPC Higgs / Z buildup
- Material choice and SEY parameters
- SEY scan and wall heat load
- Drift, dipole, quadrupole: same train

III. Outlook

- Summary
- Toward self-consistent instability simulation
- Multi-GPU with the PyPARIS logic

Buildup and Beam-Driven Instability: Where This Code Fits

Classic approach: two separate sectors

- **Buildup**: cloud formation, density, wall heat load
  - simulated with the beam train over many passages
- **Beam-driven instability**: bunch-by-bunch mode growth, emittance growth
  - a frozen cloud plus a *kick map* extracted from it

Aim of this work

- A fully self-consistent run (cloud re-solved at every bunch passage) is dominated by the field solve
- The boundary-only BEM + FMM solver is built for exactly that cost: fast enough to keep the cloud alive during multi-turn tracking
- Once optimised, the same code is intended to carry out the self-consistent instability simulation

Scope of this talk: verification of the algorithm and of the buildup physics — the foundation for the instability study that follows.

Motivation: Why a Boundary-Only Solver?

Drawbacks of PIC (PyECLOUD)

- fields live on a volume grid: charge is deposited on the mesh and fields are interpolated back
- curved walls are stair-stepped; the near-wall region needs fine grids and small time steps
- the finite-difference Poisson solver, and the scatter/gather, are hard to map onto GPUs: scatter conflicts, ghost exchange, random memory access
- $\Rightarrow$ look for a solver whose dominant cost is GPU-friendly

Boundary-only BEM + FMM

- discretize only the wall into panels; no volume Poisson mesh
- exact 2-D Green's function; arbitrary chamber cross-section
- panel normals reused for impact detection and SEY
- FMM: $O(N^2) \rightarrow O(N \log N)$; field at arbitrary points
- In practice the field is delivered on a coarse grid plus a near-wall map; the Poisson solve itself is boundary-only

BEM (1): From Poisson to a Boundary Integral

- 2-D transverse Poisson equation in the long-bunch limit:

$$\nabla^2 \phi(\mathbf{r}) = -\frac{\rho(\mathbf{r})}{\epsilon_0}, \qquad \phi = 0 \text{ on the wall } \Gamma.$$

- Free-space Green's function of the 2-D Laplacian:

$$G(\mathbf{r}, \mathbf{r}') = \frac{1}{2\pi} \ln |\mathbf{r} - \mathbf{r}'|, \qquad \nabla^2 G = \delta(\mathbf{r} - \mathbf{r}').$$

- Green's identity, with $\phi = 0$ on Γ, gives

$$\phi(\mathbf{r}) = \underbrace{-\frac{1}{2\pi\epsilon_0} \sum_k q_k \ln |\mathbf{r} - \mathbf{r}_k|}_{\phi_{\mathrm{free}}\text{: cloud charges}} + \underbrace{\frac{1}{2\pi\epsilon_0} \sum_j \sigma_j L_j \ln |\mathbf{r} - \mathbf{r}_j|}_{\phi_\sigma\text{: induced wall charges}}.$$

- The wall is split into N panels of length L_j; panel j carries the line charge $\sigma_j L_j$, i.e. σ_j is the charge per unit length of that panel.

BEM (2): Collocation Turns Poisson into a Matrix

- Enforce $\phi = 0$ at the midpoint of every panel i:

$$\phi_{\text{free}}(\mathbf{r}_i) + \phi_\sigma(\mathbf{r}_i) = 0 \quad \Longrightarrow \quad \sum_{j=1}^{N} A_{ij}\sigma_j = b_i.$$

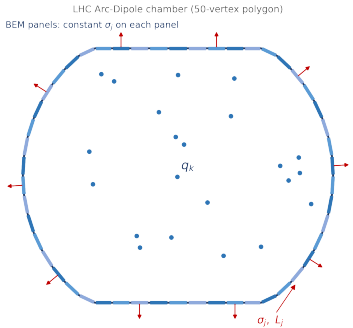
- Matrix elements (2-D log kernel):

$$A_{ij} = -\frac{L_j}{2\pi} \ln r_{ij} \quad (i \neq j),$$

$$A_{ii} = \frac{L_i}{2\pi}\left(1 - \ln \frac{L_i}{2}\right),$$

$$b_i = \frac{1}{2\pi} \sum_k q_k \ln r_{ik}.$$

- A depends *only on the geometry*: LU factorization once, reused for every source update.



LHC arc-dipole chamber (50-vertex polygon); σ_j: charge per unit length on panel j.

BEM (3): Gauss-Law Correction and Cost

- For a closed conducting chamber the induced charge must cancel the source charge:

$$\sum_{j=1}^{N} \sigma_j L_j = -\sum_k q_k.$$

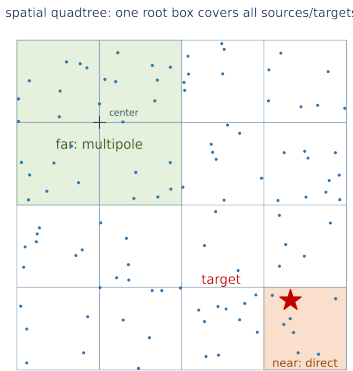
- After the solve, apply a uniform correction:

$$\alpha = \frac{-\sum_k q_k - \sum_j \sigma_j L_j}{\sum_j L_j}, \qquad \sigma_j \leftarrow \sigma_j + \alpha.$$

- A uniform α shifts the potential by a constant inside the chamber, so the field is unchanged; it removes the constant (null) mode.
- In this talk: $N = 4096$ panels.
  - LU factorization once, reused while the geometry is unchanged
  - each source update is an $O(N^2)$ triangular solve with $N \sim$ a few thousand $\Rightarrow$ about 10 ms, negligible

FMM (1): Tree Code + Multipole Expansion

- Direct summation costs $O(N_t N_s)$ per field evaluation — too slow for 10^5–10^6 particles
- Idea: group distant sources into boxes, and replace the box by its multipole expansion about the box centre
- A quadtree subdivides space recursively
- Near boxes: direct sum; far boxes: multipole/local expansion
- Accuracy set by the opening angle θ and the order p; cost reduces to $O\big((N_t + N_s)\log(N_t + N_s)\big)$



FMM (2): Upward Pass — P2M and M2M

Write $z = x + iy$. The 2-D kernel is $\mathrm{Re}\,[q \ln(z_j - z_i)]$; all expansions are taken about the box center, order $p = 10$.

P2M (particles $\to$ multipole of box s):

$$M_n^{(s)} = \sum_{i \in s} q_i \, (z_i - z_s)^n, \qquad n = 0, \ldots, p$$

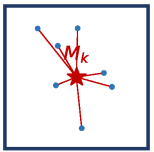
- M_0 is the total charge of the box
- one pass over the particles, no pair interaction

M2M (child $c \to$ parent p, $\delta = z_c - z_p$):

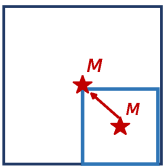
$$M_n^{(p)} = \sum_{k=0}^{n} \binom{n}{k} \delta^{\,n-k} M_k^{(c)}$$

- re-centres the expansion, never touches the particles
- cost $O(p^2)$ per box, once per level

P2M: particles → multipole



M2M: child → parent



The upward pass only: P2M at the leaves, then M2M level by level up to the root.

FMM (3): Downward Pass and Near Field

M2L (multipole → local, $d = z_t - z_s$):

$$L_0 = M_0 \ln d - \sum_{n \geq 1} \frac{M_n}{n\, d^{\,n}}$$

$$L_k = \frac{1}{d^{\,k}} \left[\frac{(-1)^{k+1} M_0}{k} - (-1)^k \sum_{n=1}^{p-k} \frac{M_n}{n} \binom{n+k-1}{k} \frac{1}{d^{\,n}} \right]$$

$k = 1, \ldots, p$; applied once per box pair on the interaction list.

L2L (parent $p \to$ child c, $\delta = z_c - z_p$):

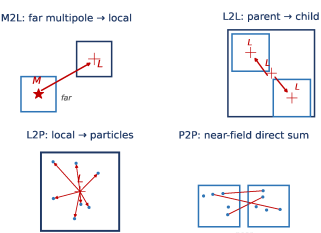
$$L_n^{(c)} = \sum_{k=n}^{p} \binom{k}{n} \delta^{\,k-n} L_k^{(p)}$$

L2P (local → particles in box t):

$$\phi(z_j) = \mathrm{Re} \sum_{k=0}^{p} L_k^{(t)} (z_j - z_t)^k, \quad \mathbf{E} = -\nabla \phi$$

P2P (near neighbours, direct):

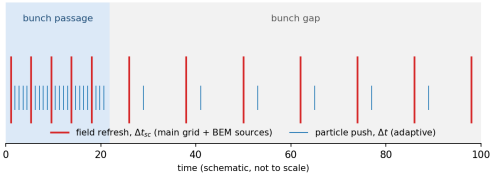
$$\phi_j = \sum_{i \in \mathcal{N}(j)} q_i \ln |z_j - z_i|$$



$\theta = 0.7$, $p = 10$, error $\sim 10^{-5}$, total $O(N \log N)$.

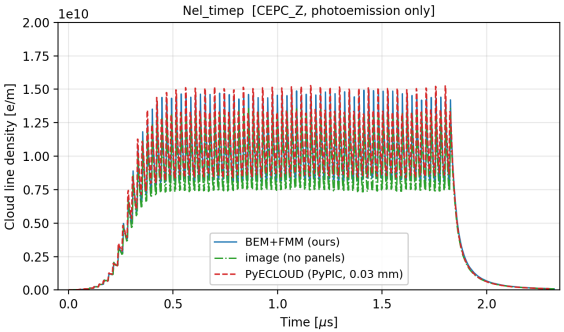
Coupling the Field Solver to the Tracking Loop

- Field seen by the electrons:
  $\phi = \phi_{\mathrm{free}}(\text{cloud} + \text{beam}) + \phi_\sigma(\text{wall})$
- One unified source set — cloud charges q_k and wall charges $\sigma_j L_j$ — evaluated by the FMM in one pass
- Field refreshed on a main grid and interpolated to the particles; near the wall, a fine polar map or direct evaluation
- Geometry change $\rightarrow$ rebuild panels and LU only
- The boundary solve can also feed an FFT-based PIC solver



- `BorisMultipole`: Boris pusher in the multipole field of the chamber magnets
- Two nested steps: field refresh Δt_{sc}, particle push Δt (adaptive; impacts checked at every substep)
- Beam field added linearly; drop-in SC backend for PyECLOUD

Validation: BEM vs Analytic Image vs PyECLOUD



CEPC Z: cloud line density — BEM, image and PyECLOUD, same seed and SEY.

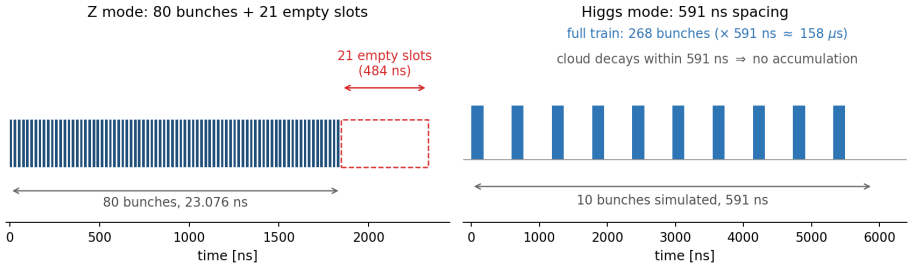
Quantity	BEM+FMM vs image	PyECLOUD vs image
N_{el}	+10%	+17%
E_{kin}	+8%	−14%

- Reference: exact 2-D image of the circular chamber, $r' = (R^2/r^2)r$, $q' = -q$
- N_{el}: line density averaged over the 80 bunch passages (time-weighted mean)
- All three use the same SEY: ECLOUD/Hilleret with the $E > 500$ eV correction
- BEM+FMM follows the image to 10% in N_{el} and 8% in E_{kin}; the residual is generated in the exponential buildup, not at saturation
- BEM and PyECLOUD agree with each other to 7%

CEPC Buildup: Simulation Setup

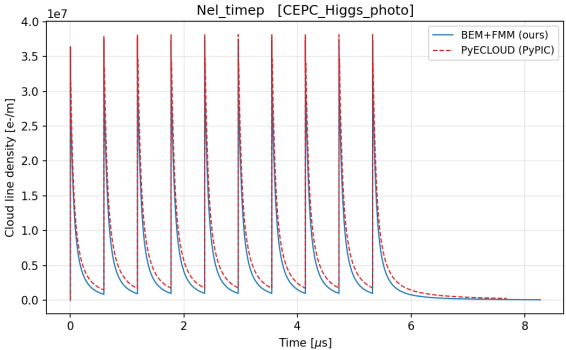
Parameter	Z mode	Higgs mode
Beam energy	45.5 GeV	120 GeV
Bunch population	1.4×10^{11}	1.3×10^{11}
Bunch spacing	23.076 ns	591 ns
Arc dipole field	14.2 mT	37.4 mT
Chamber	circular, 56 mm	circular, 56 mm

- Secondary electrons: ECLOUD/Hilleret ($\delta_{\max} = 1.9$, $E_{\max} = 332$ eV, $R_0 = 0.7$)
- Photoemission with reflection
- `BorisMultipole` in the arc dipole
- Adaptive step: fine near the bunch, coarse in the gap
- In this talk: $N = 4096$ panels



Z: one 80-bunch train + 21 empty slots. Higgs: 10 of the 268 bunches — the cloud decays within 591 ns.

CEPC Higgs Mode: Transient Cloud, No Accumulation



Higgs with photoemission: cloud line density.

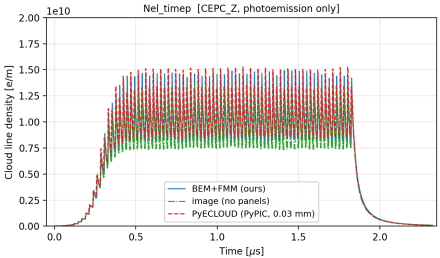
- Each bunch produces a transient cloud
  - peak $N_{el} \approx 3.8 \times 10^7$ e/m
  - peak central density $\approx 1.0 \times 10^{10}$ m^{-3}
- Agrees with PyECLOUD to $\sim 2\%$ (N_{el}) and $\sim 0.1\%$ (central density)
- Decays 2–3 orders within 591 ns: no accumulation, so 10 bunches are enough
- Peak ~ 2.5 orders below Z

Photoelectrons per unit length, per bunch:

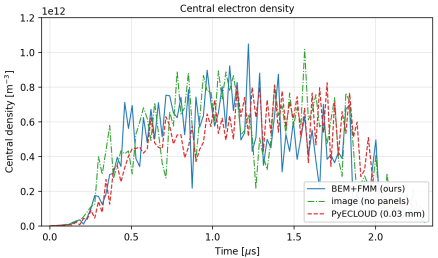
$$\frac{dN_{pe}}{dt} = k_{pe,st}\, c\, \lambda_b(t) \implies \frac{N_{pe}}{L} = k_{pe,st}\, N_b$$

$k_{pe,st}$ [m^{-1}]: photoelectrons per metre per bunch.

CEPC Z Mode: Cloud Buildup and Saturation



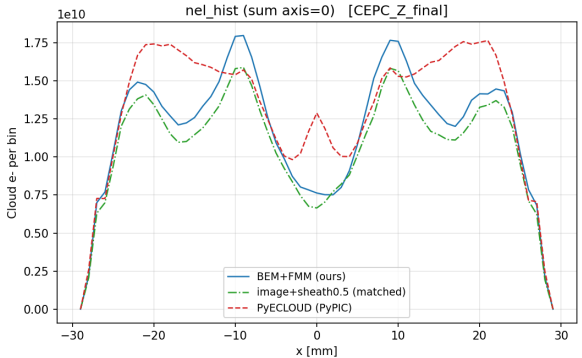
Line density.



Central density.

- Builds up over the train and saturates at $\delta_{\text{eff}} = 1.00 \pm 0.01$
- Train-mean line density $\sim 1 \times 10^{10}$ e/m; peaks 1.5–1.8 $\times 10^{10}$ e/m
- Central density rises from 3×10^{9} to $\sim 8 \times 10^{11}$ m^{-3}; both codes follow the same curve

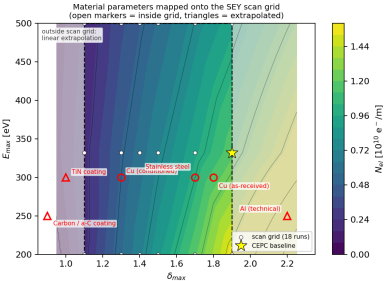
CEPC Z Mode: Transverse Cloud Structure

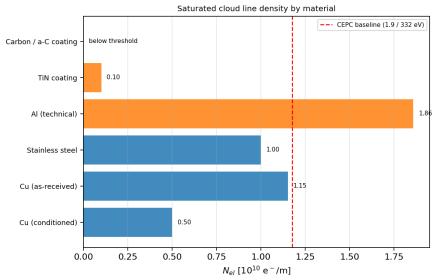


Cloud transverse profile (sum over y), 14.2 mT arc dipole.

- BEM and analytic image agree on a central depletion
- Peaks near $x \approx \pm 10$ mm and ± 20 mm
- Impacts concentrate on the top/bottom and side walls
- PyECLOUD shows a central bump: its 0.5 mm grid fills in the near-axis region
- The transverse profile matters for CBI, which samples the cloud near the beam axis

Material Choice and the SEY Parameters Used



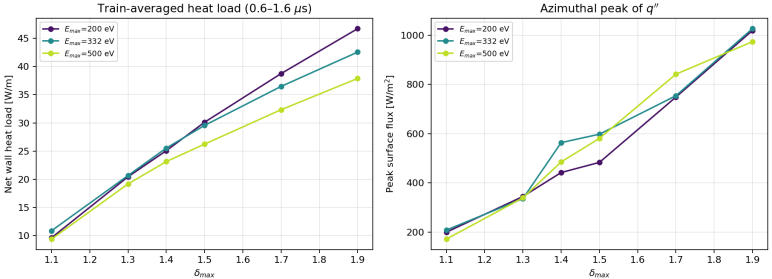


$$\delta(E) = \delta_{\max} \frac{s\,x}{s - 1 + x^s} + R_0 \left(\frac{\sqrt{E} - \sqrt{E + E_0}}{\sqrt{E} + \sqrt{E + E_0}} \right)^2$$

$$x = \frac{E}{E_{\max}}, \quad s = 1.35, \quad E_0 = 150\ \text{eV}, \quad R_0 = 0.7$$

- EDR baseline: treated Cu, $\delta_{\max} = 1.9$, $E_{\max} = 332$ eV, $R_0 = 0.7$
- Surface treatment matters more than the field map: treated vs untreated Cu gives 2.3×
- TiN or carbon coating: about one order of magnitude lower
- Placeholder literature values, used as effective parameters

SEY Scan: Cloud Density and Wall Heat Load

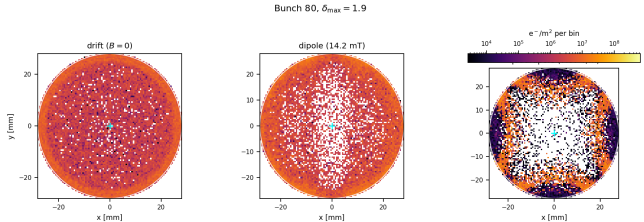


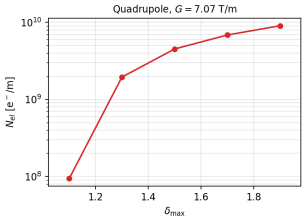
Net wall heat load and azimuthal peak versus $\delta_{\max}$, for $E_{\max} = 200/332/500$ eV.

- 18 runs: $\delta_{\max} = 1.1$–1.9, $E_{\max} = 200/332/500$ eV; all saturate at $\delta_{\text{eff}} = 1.00 \pm 0.01$
- $\delta_{\max}$ 1.9→1.1: cloud $\times 0.18$, heat load $\times 0.26$ (the surviving electrons hit harder)
- $E_{\max}$ 200→500 eV: cloud -25–40%, heat load only -5–20%

- Baseline (1.9/332): 42.6 W/m in the train, 28.5 W/m cycle-averaged, peak 0.10 W/cm^2
- 28.5 W/m is 1–1.5 orders above a $\sim$1 W/m cold-bore budget
- $\Rightarrow$ CEPC needs a low-SEY surface: a TiN coating, not bare copper

Same Train, Three Field Configurations





Quadrupole: saturated N_{el} versus $\delta_{\max}$.

- Build-up is almost field-independent: both the saturation level (9.4 / 11.7 / 10.7 $\times$ 10^9 e/m) and the $\sim$0.5 μs rise are the same in drift, dipole and quadrupole
- The field only redistributes the cloud: central density 2.3 $\times$ 10^{12} / 1.0 $\times$ 10^{12} / 3.1 $\times$ 10^9 m^{-3} (2.3$\times$ / 740$\times$ lower), $\langle E_{\mathrm{imp}} \rangle$ 452 / 212 / 280 eV
- Quadrupole only: N_{el} grows $10^8 \rightarrow 10^{10}$ e/m over $\delta_{\max} = 1.1 \rightarrow 1.9$ and saturates above 1.5
- Azimuthal contrast 0.98 / 0.79 / 26: only the quadrupole keeps structure ($x \cdot y$ field); its fine striping is sampling-dependent (2 ns $\rightarrow$ 10 ps: 3.4 $\times$ 10^3 $\rightarrow$ 50)

Summary and Outlook

- Boundary-only BEM + FMM solver (no volume Poisson mesh): dense matrix $A\sigma = b$, tree code $O(N \log N)$, GPU-accelerated
- Validation against the analytic circular-chamber image: N_{el} +10%, central density +2%, mean impact energy −1%, E_{kin} +8%
- CEPC results:
  - Z mode: $\sim 1.2 \times 10^{10}$ e/m, $\tau \approx 100$ ns, bunch-to-bunch accumulation; 28.5 W/m cycle-averaged
  - Higgs mode: transient cloud ($\sim 3.8 \times 10^{7}$ e/m), no accumulation; heat load ∼5 orders lower
  - Quadrupole at 7.07 T/m: 10^{8}–10^{10} e/m over $\delta_{\max} = 1.1$–1.9, saturating above 1.5; same total cloud as the dipole, central density 740× lower
- Outlook: self-consistent CBI / emittance growth with the same solver (no kick map)
- Multi-GPU with the PyPARIS logic: bunches spread over ranks and advanced concurrently, so the wall-clock follows one bunch passage

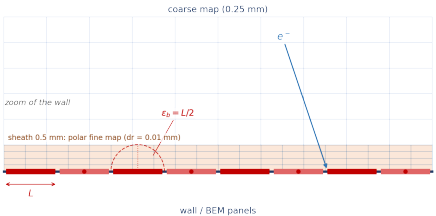
Single-case baseline: 601 s (1 CPU) vs 379 s (1 GPU); multi-GPU not yet exploited.

Backup: Near-Wall Treatment (not presented)

- A uniform map ($dh = 0.25$ mm) cannot resolve the < 0.1 mm boundary layer
- Sheath: electrons within 0.5 mm of the wall bypass the coarse map
  - direct field evaluation, or a polar fine map
- Boundary-source softening:

$$\epsilon_b = \frac{L}{2} = \frac{\pi R}{N} = 0.0215 \text{ mm} \quad (N = 4096)$$

  - avoids a spurious $1/d$ singularity ($\sim 10^6$ V/m)
- Polar sheath map: $dr = 0.01$ mm, $n_\phi = 8192$, rebuilt every 0.5 ns
- Interpolation error vs direct evaluation: $< 0.1\%$ beyond 0.05 mm



Boundary-field error vs image: $< 1\%$ beyond 0.1 mm; the < 0.05 mm layer carries only $\sim$3.5% of the charge.

Backup: Performance (not presented)

Configuration	Main-loop time	Remark
PyECLOUD 8.7.1 (PyPIC, 1 core)	∼601 s (10.0 min)	reference
BEM 512 + sheath (coarse)	502 s (8.4 min)	accuracy insufficient
BEM 4096 + sheath, direct field	764 s (12.7 min)	accurate, slower
BEM 4096 + sheath, fine map	379 s (6.3 min)	production, 1.6× faster
BEM 8192 + sheath, direct	1158 s (19.3 min)	no gain, high-energy tail

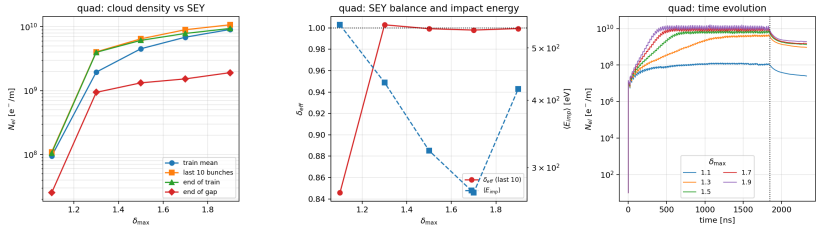
Time breakdown (379 s)	Time	Share
`sc_map` (main + sheath map)	251 s	66%
`sc_gather` (substep interpolation)	62 s	16%
beam / Boris / SEY / recording	∼66 s	17%

Current single-case baseline: 1 CPU core vs 1 GPU. Multi-GPU and multi-case concurrency are not yet exploited; the bottleneck is per-particle work and host–device copies, not the field solve.

Backup: CEPC Z-Mode CBI (in progress)

- Stage 1 (kick map): use the steady-state cloud of one Z-mode train as a frozen background
- Multi-turn tracking to extract bunch-by-bunch transverse mode growth rates
- Cross-check against the frequency-domain model and PyPARIS
- Stage 2 (target): drop the frozen cloud and the kick map; re-solve the cloud at every bunch passage with the BEM + FMM solver
- Focus: how the 14.2 mT field and the transverse cloud profile affect CBI
- Status: stage 1 in progress; stage 2 follows the solver optimisation

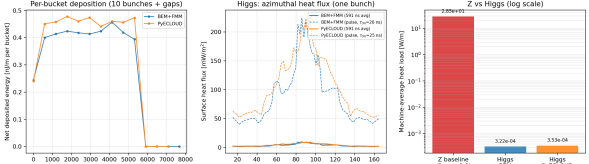
Backup: Quadrupole SEY Scan (not presented)



$G = 7.07$ T/m, 80 bunches + 21 gap slots, high-energy $\delta(E)$ correction above 500 eV.

- N_{el} from 10^8 ($\delta_{\max} = 1.1$) to 10^{10} e/m ($\delta_{\max} = 1.9$), saturated above 1.5
- $\delta_{\text{eff}} \to 1.00$ for $\delta_{\max} \geq 1.3$
- The absolute value at 1.9 depends on the high-energy part of $\delta(E)$: treat it as a conservative estimate

Backup: Higgs Wall Heat Load (not presented)



Higgs wall energy deposition and heat load (BEM vs PyECLOUD).

- 4.01×10^{-10} J/m deposited per bunch
- Machine average 3.22×10^{-4} W/m
- $\sim$5 orders below the Z baseline (28.5 W/m)
- 90% of the energy is deposited within $\sim$25 ns
- No heat-load or quench risk for either chamber type