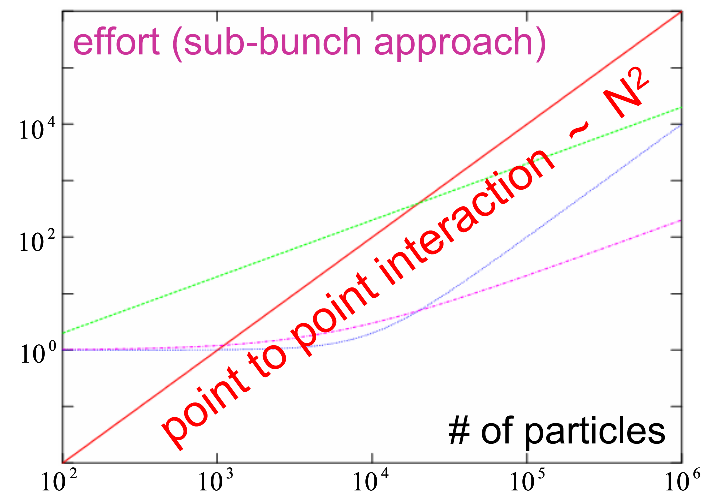
CSRtrack: Faster Calculation of 3d CSR Effects

M. Dohlus, T. Limberg, DESY, Hamburg, Germany

- Projected Method / Sub-Bunch Approach
- Field Calculation: Convolution Method
  Pseudo Green's Function Approach
  Meshed Fields
- Iterative Tracking
- CSRtrack
- Example

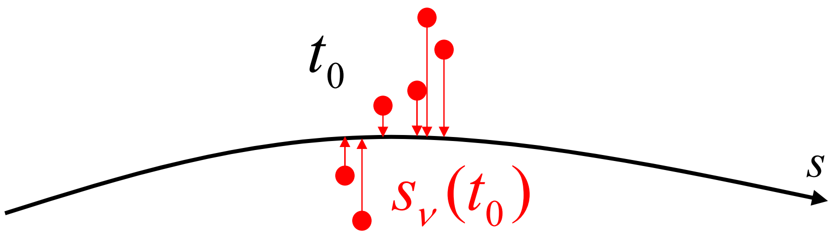


Projected Method

(1d approach)

$$\dot{\mathbf{p}}_\nu = q\left(\mathbf{e}_{\nu\|} E^{(\lambda)}(s_\nu, t) + \mathbf{v}_\nu \times \mathbf{B}^{(\text{ext})}\right)$$

no transverse self-forces



rigid 1d charge distribution:

$$\lambda^{(\delta)}(s - t_0 c) = \sum q_\nu \delta((s - t_0 c) - (s_\nu - s))$$

$$\lambda(s - t_0 c) = \lambda^{(\delta)}(s - t_0 c) \otimes \left(g\left(s/\sigma\right)/\sigma\right)$$

1d E-field without γ^{-2} singularity:

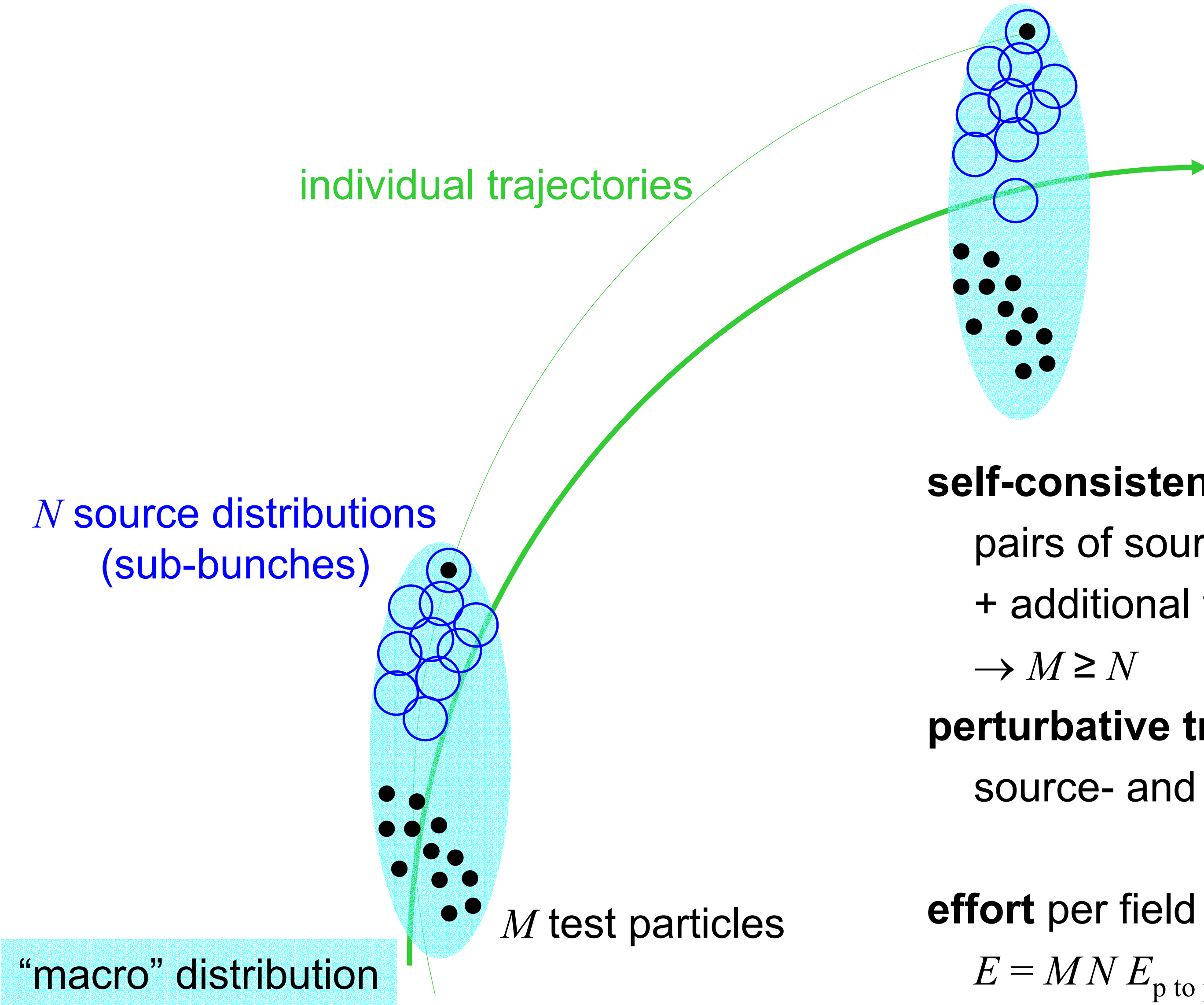
$$E^{(\lambda)}(s, t_0) = \int \lambda'(u + s - ct_0) K(s, u) du$$

no transverse dependency
of longitudinal forces

very low numerical effort

see: Saldin, Schneidmiller, Yurkov: NIM A398 (1997) 373-394
Dohlus: TESLA-FEL-2003-05

Sub-Bunch Approach



self-consistent tracking:

pairs of source- and test-particles

+ additional test particles

$\rightarrow M \geq N$

perturbative tracking:

source- and test-particles independent

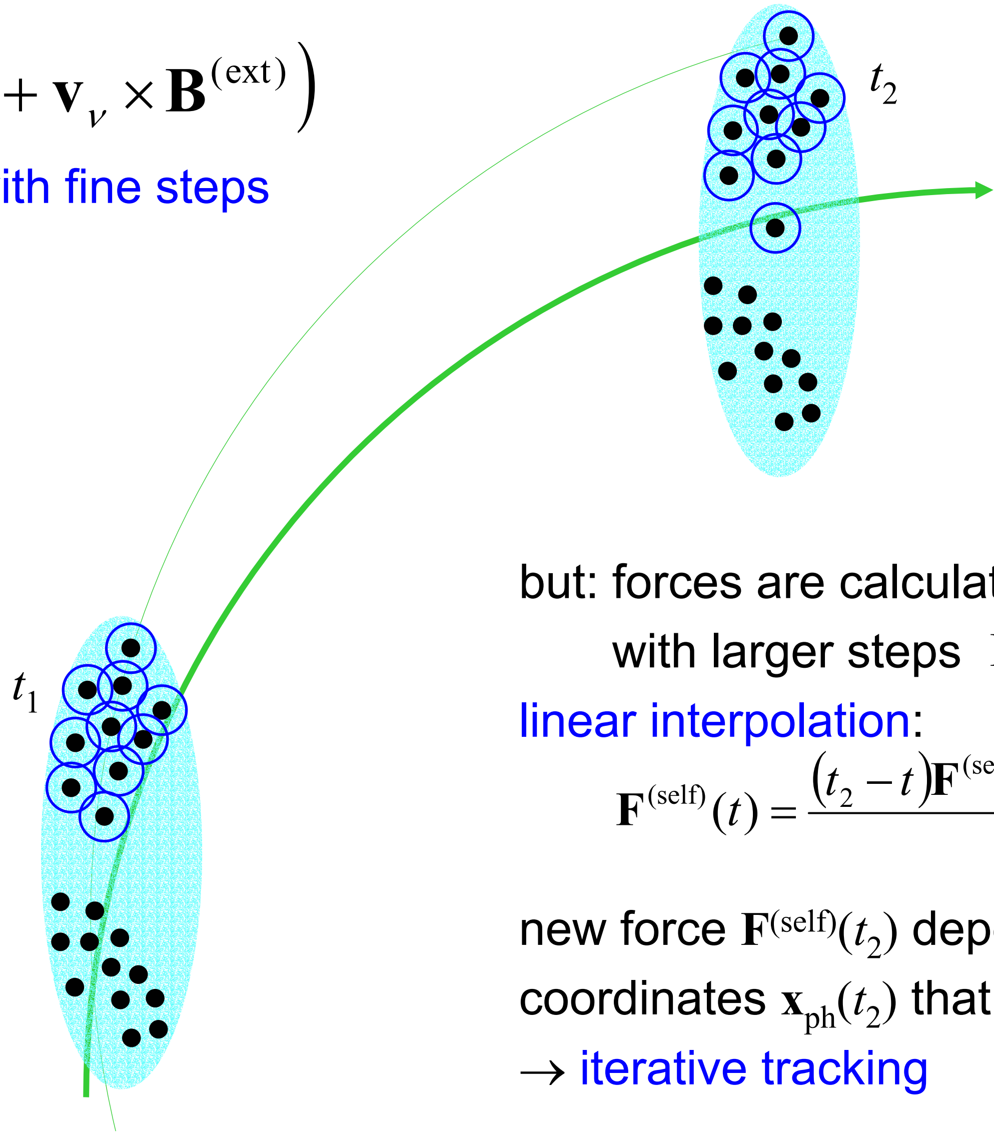
effort per field calculation:

$E = M\,N\,E_{\mathrm{p\ to\ p}}$

Iterative Tracking

$$\dot{\mathbf{p}}_\nu = q\left(\mathbf{F}_\nu^{(\mathrm{self})} + \mathbf{v}_\nu \times \mathbf{B}^{(\mathrm{ext})}\right)$$

integration with fine steps



but: forces are calculated on time mesh

with larger steps $\mathbf{F}^{(\mathrm{self})}(t_1), \mathbf{F}^{(\mathrm{self})}(t_2), \cdots$

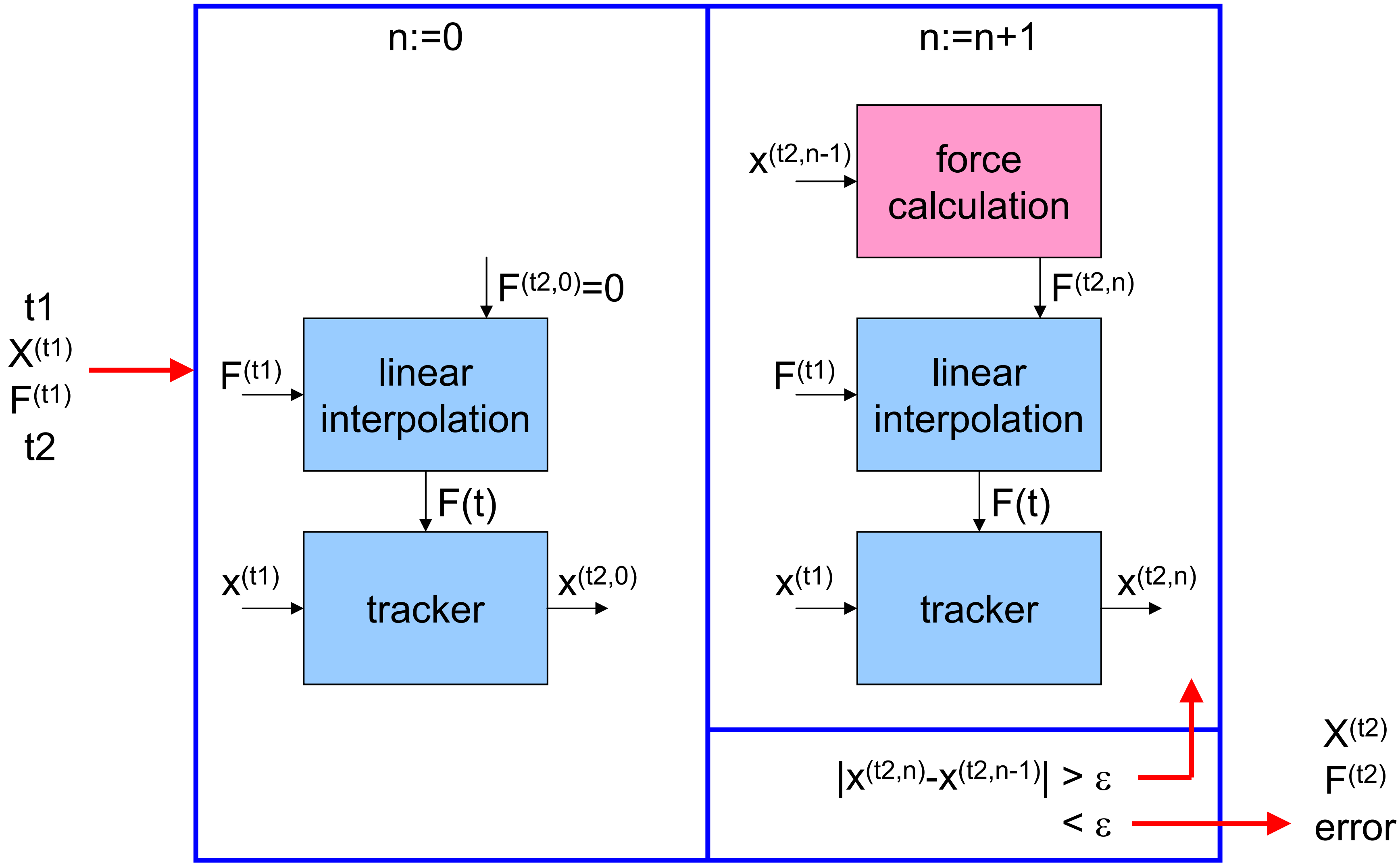
linear interpolation:

$$\mathbf{F}^{(\mathrm{self})}(t) = \frac{(t_2 - t)\mathbf{F}^{(\mathrm{self})}(t_1) + (t - t_1)\mathbf{F}^{(\mathrm{self})}(t_2)}{t_2 - t_1}$$

new force $\mathbf{F}^{(\mathrm{self})}(t_2)$ depends on new phase space coordinates $\mathbf{x}_{\mathrm{ph}}(t_2)$ that depend on $\mathbf{F}^{(\mathrm{self})}(t_2)$

$\rightarrow$ iterative tracking

Iterative Tracking

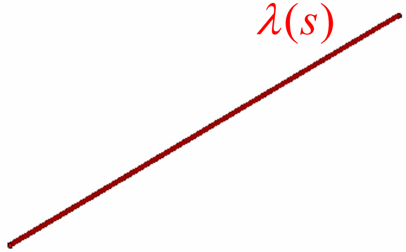


Convolution Method

source distributions:

1d distribution

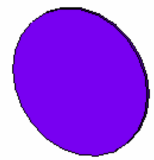
$\lambda(s)$



$\otimes$

2d distribution

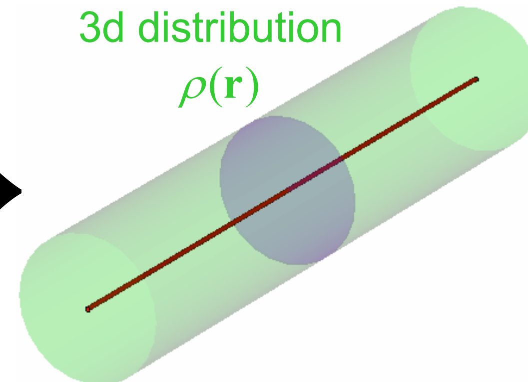
$\eta(r,z)$



$\rightarrow$

3d distribution

$\rho(\mathbf{r})$



em fields:

$$\mathbf{E}^{(\lambda)}(\mathbf{r}), \mathbf{B}^{(\lambda)}(\mathbf{r}), \quad \otimes \quad \eta \quad \rightarrow \quad \mathbf{E}^{(3d)}(\mathbf{r}), \mathbf{B}^{(3d)}(\mathbf{r}),$$

singular (s) and non-singular (ns) parts:

$$\mathbf{E}^{(\lambda)}(\mathbf{r}) = \mathbf{E}_{s}^{(\lambda)}(\mathbf{r}) + \mathbf{E}_{ns}^{(\lambda)}(\mathbf{r})$$

$$\mathbf{B}^{(\lambda)}(\mathbf{r}) = \ldots$$

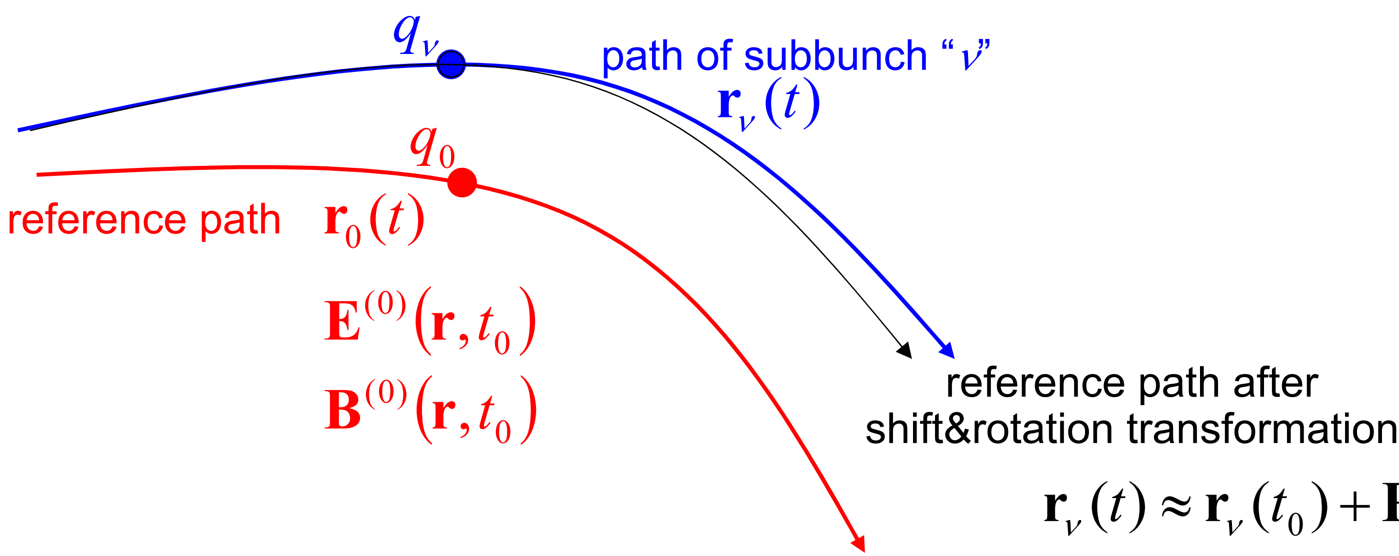
$$\mathbf{E}^{(3d)}(\mathbf{r}) = \eta \otimes \mathbf{E}_{s}^{(\lambda)}(\mathbf{r}) + \eta \otimes \mathbf{E}_{ns}^{(\lambda)}(\mathbf{r})$$

$$\approx \boxed{\eta \otimes \mathbf{E}_{s}^{(\lambda)}(\mathbf{r})} + \boxed{\mathbf{E}_{ns}^{(\lambda)}(\mathbf{r})}$$

analytical functions | numerical 1d integration

see: M.Dohlus, A.Kabel, T.Limberg: Efficient field calculation of 3D bunches on general trajectories. NIM A445 (2000) 338-342

Pseudo Green's Function Approach



$$\mathbf{r}_\nu(t) \approx \mathbf{r}_\nu(t_0) + \mathbf{R}\cdot\left(\mathbf{r}_0(t) - \mathbf{r}_0(t_0)\right)$$

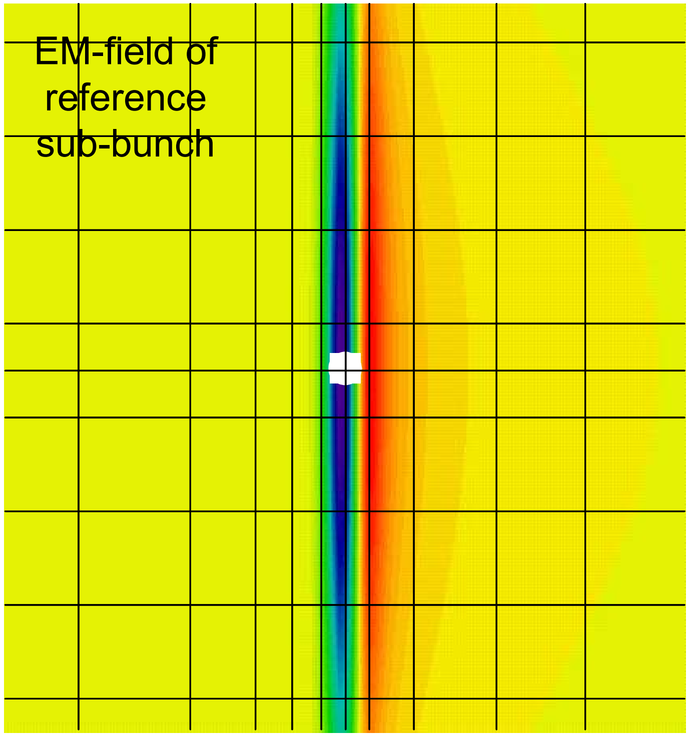
$$\mathbf{E}^{(\nu)}(\mathbf{r},t_0) \approx \frac{q_\nu}{q_0}\mathbf{R}\cdot\mathbf{E}^{(0)}\left(\mathbf{r}_0(t_0) + \mathbf{R}^{-1}(\mathbf{r} - \mathbf{r}_\nu(t_0)), t_0\right)$$

$$\mathbf{B}^{(\nu)}(\mathbf{r},t_0) \approx \frac{q_\nu}{q_0}\mathbf{R}\cdot\mathbf{B}^{(0)}\left(\mathbf{r}_0(t_0) + \mathbf{R}^{-1}(\mathbf{r} - \mathbf{r}_\nu(t_0)), t_0\right)$$

the “Green’s functions” are calculated numerically on a 2d-mesh with M_{g} points

$$\mathbf{E}^{(0)} = E_x^{(0,t_0)}(x,y)\mathbf{u}_x + E_y^{(0,t_0)}(x,y)\mathbf{u}_y$$

$$\mathbf{B}^{(0)} = B_z^{(0,t_0)}(x,y)\mathbf{u}_z$$



effort per field calculation:

$$E_{f.c.} = M_g E_{\mathrm{p\,to\,p}} + NME_{\mathrm{i,g}}$$

$E_{\mathrm{p\,to\,p}}$ = effort per point to point interaction

$E_{\mathrm{i,g}}$ = effort for interpolation on grid << $E_{\mathrm{p\,to\,p}}$

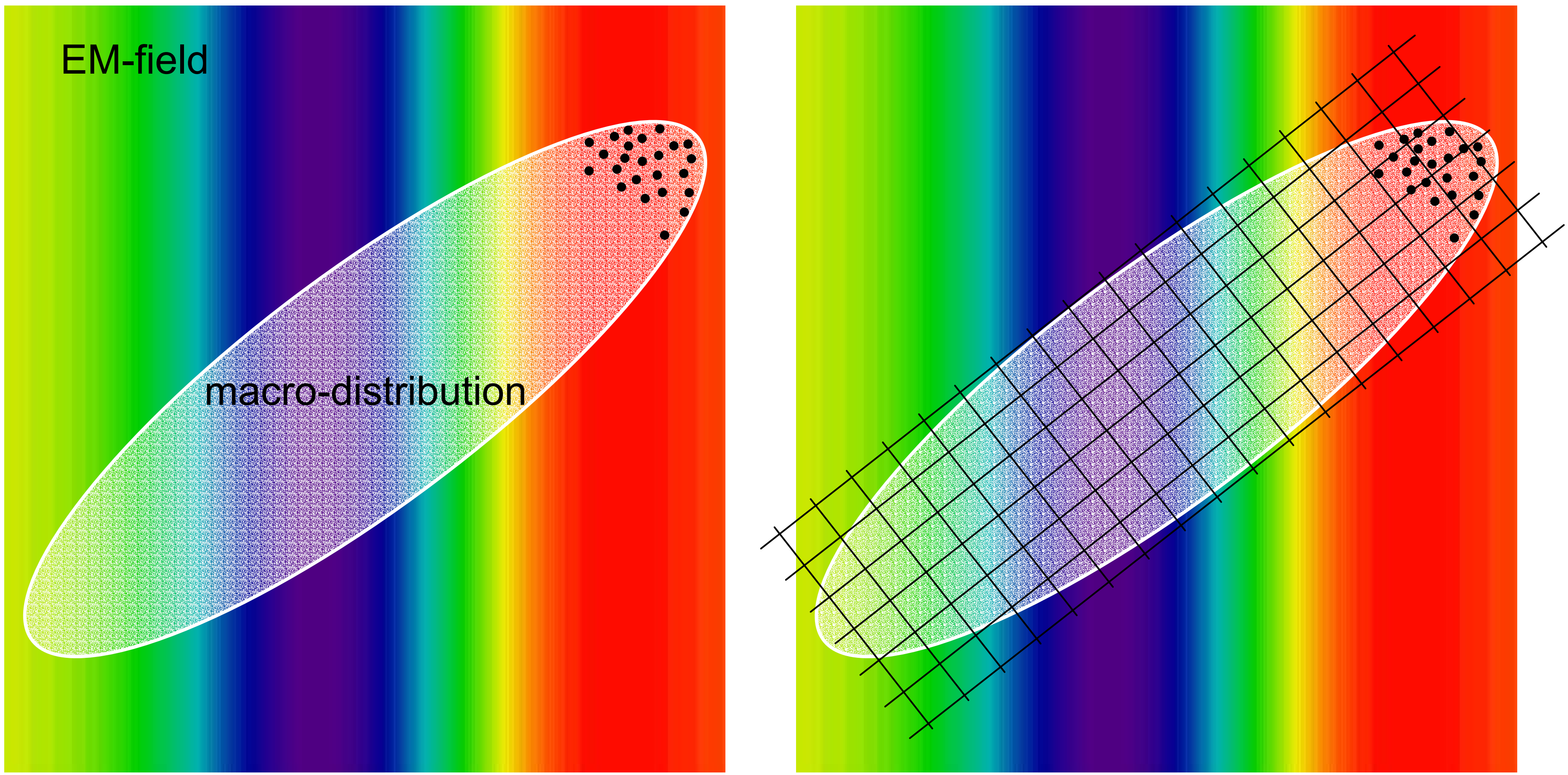
N = number of source distributions

M = number of test particles ($> N$)

Meshed EM Fields

not now in CSRtrack

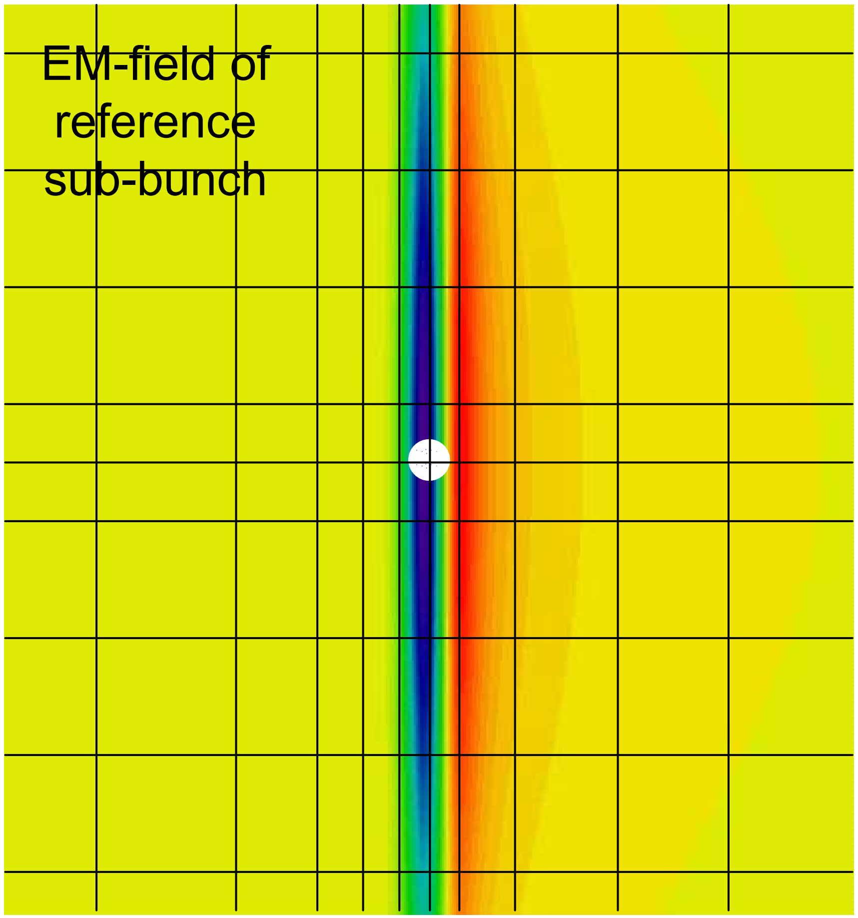
density of particles large compared to fine structure of field:
calculate field on mesh (M_{em} points) and interpolate to M test-particles
effort: $E_{\text{f.c.}} = M_{\text{em}} N E_{\text{p to p}} + M\, E_{\text{i,em}}$ (~ linear with particles)

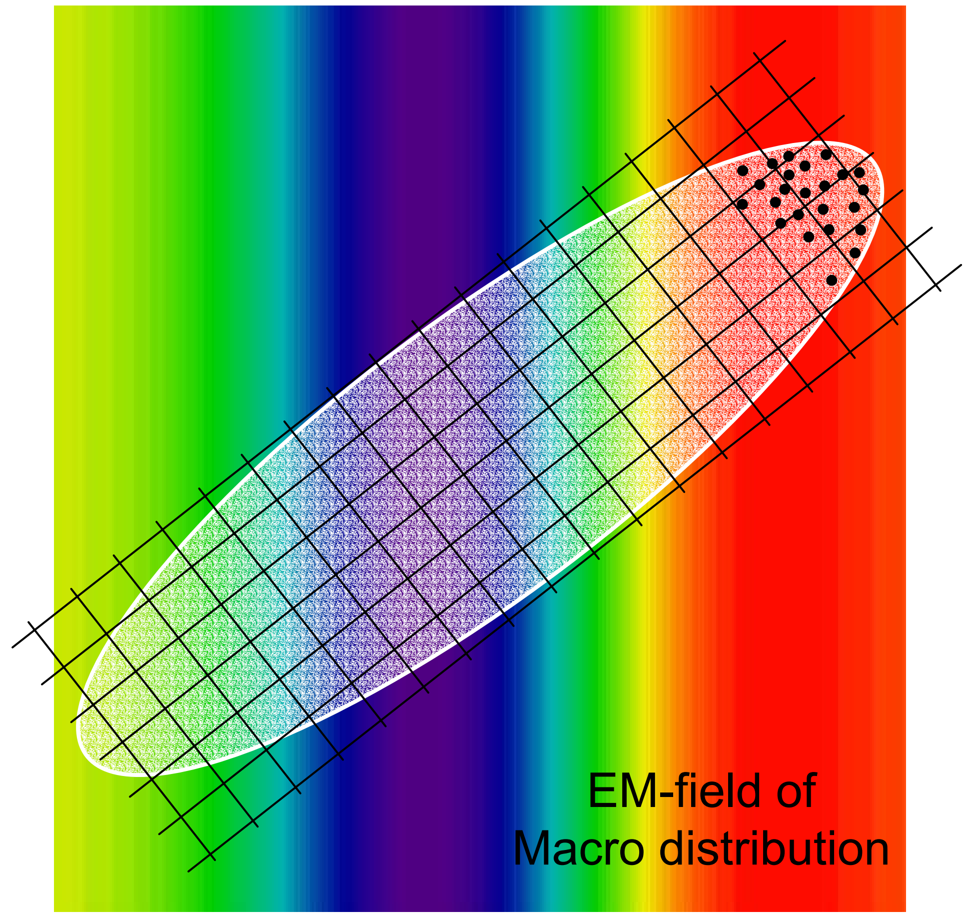


Meshed EM Fields + Pseudo Green's ...

not now in CSRtrack

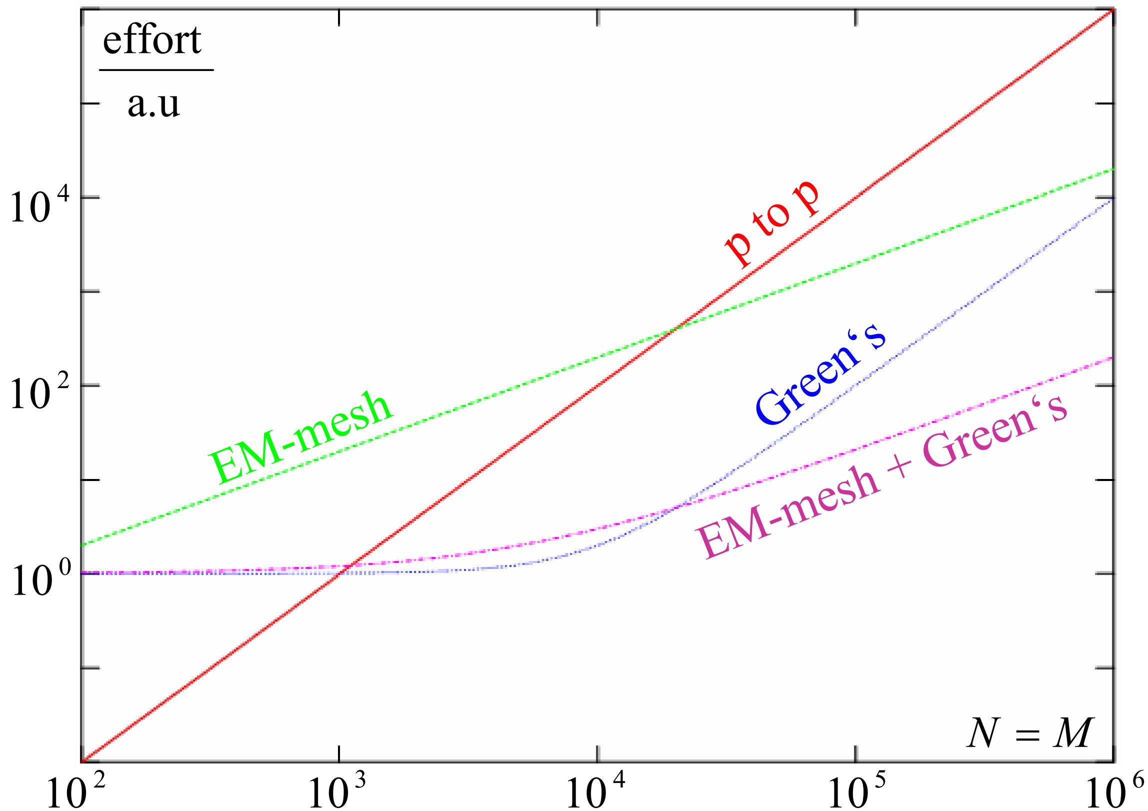
effort: $E_{\text{f.c.}} = M_g E_{\text{p to p}} + M_{\text{em}} N E_{\text{i,g}} + M\, E_{\text{i,em}}$

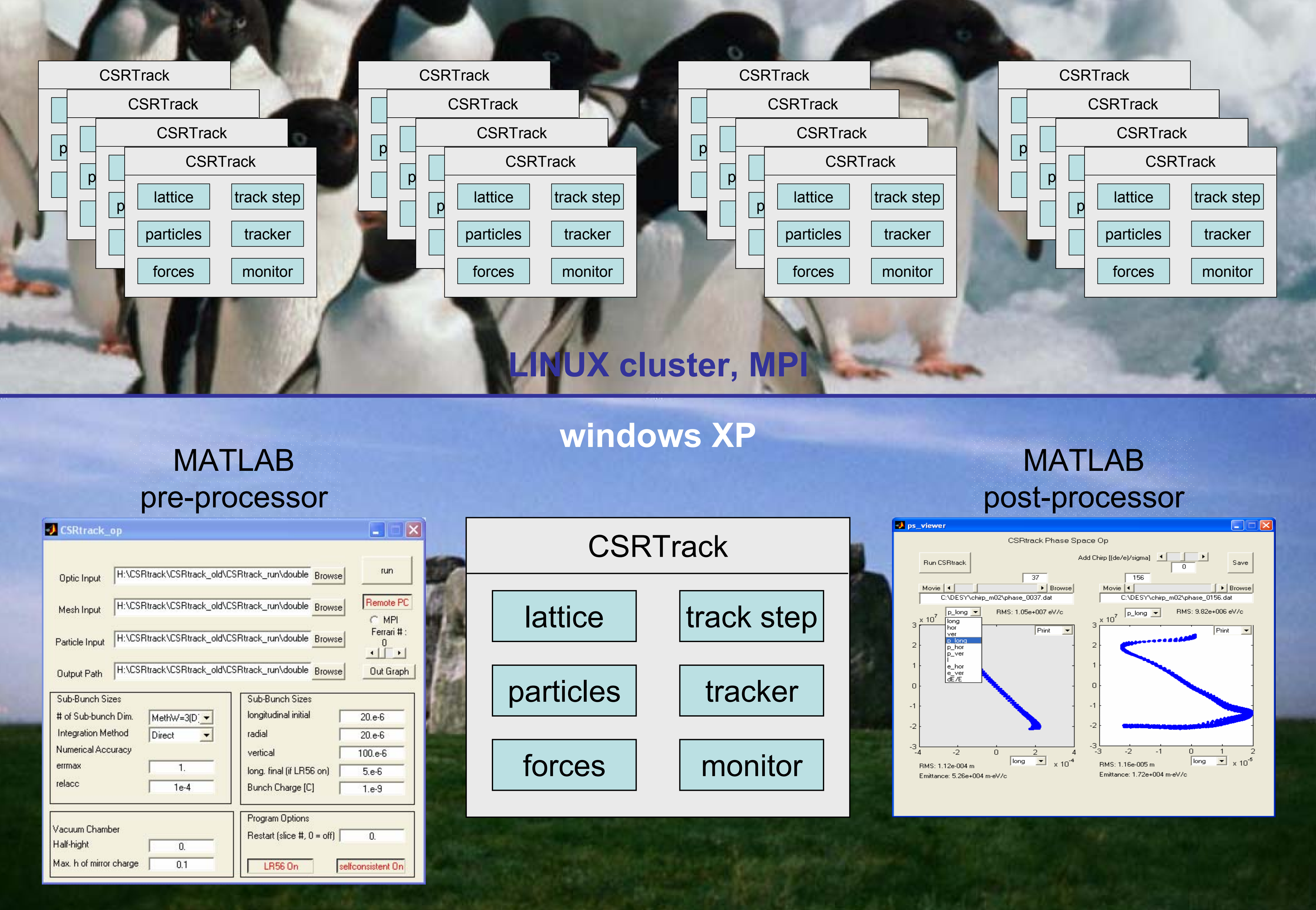




Scaling of Effort

(simplified)



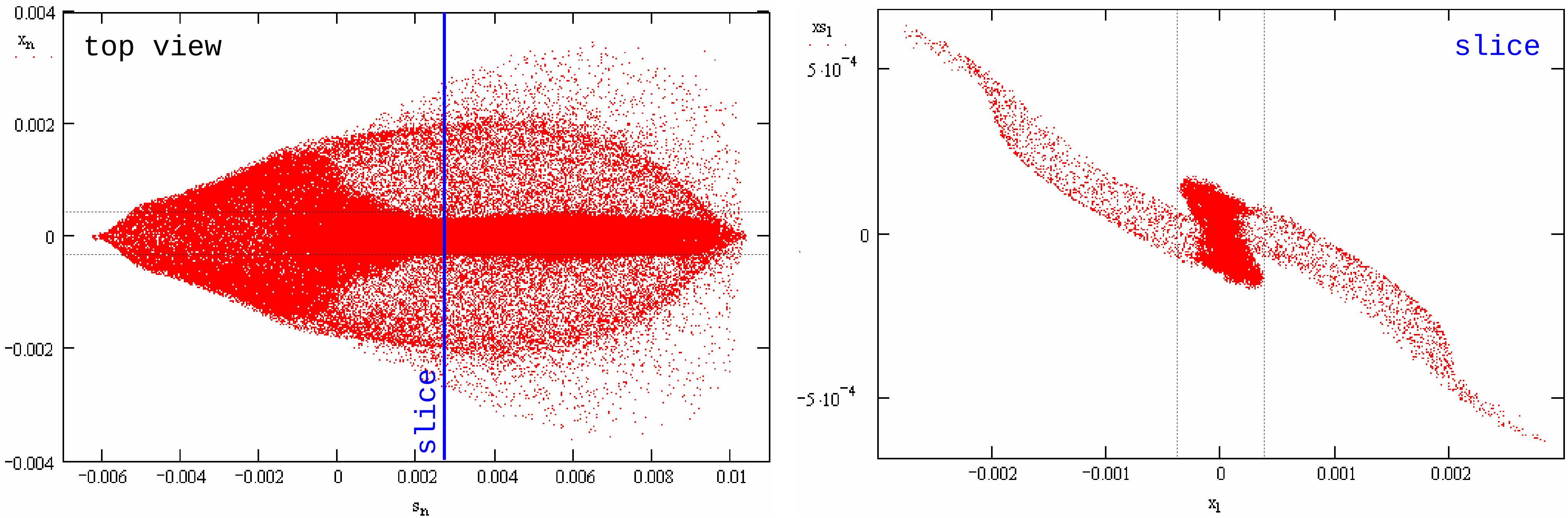

CSRTrack
lattice
track step
particles
tracker
forces
monitor
LINUX cluster, MPI
windows XP
MATLAB
pre-processor
MATLAB
post-processor
CSRtrack_op
Optic Input
Mesh Input
Particle Input
Output Path
H:\CSRtrack\CSRtrack_old\CSRtrack_run\double
Browse
run
Remote PC
MPI
Ferrari # :
Out Graph
Sub-Bunch Sizes
# of Sub-bunch Dim.
Integration Method
Direct
Numerical Accuracy
errmax
1.
relacc
1e-4
longitudinal initial
20.e-6
radial
20.e-6
vertical
100.e-6
long. final (if LR56 on)
5.e-6
Bunch Charge [C]
1.e-9
Vacuum Chamber
Half-hight
0.
Max. h of mirror charge
0.1
Program Options
Restart (slice #, 0 = off)
0.
LR56 On
selfconsistent On
ps_viewer
CSRtrack Phase Space Op
Run CSRtrack
Add Chirp [(de/e)/sigma]
Save
Movie
Browse
C:\DESY\chirp_m02\phase_0037.dat
C:\DESY\chirp_m02\phase_0156.dat
RMS: 1.05e+007 eV/c
RMS: 9.82e+006 eV/c
Print
RMS: 1.12e-004 m
Emittance: 5.26e+004 m-eV/c
RMS: 1.16e-005 m
Emittance: 1.72e+004 m-eV/c

Example: BC2-TTF1

overcompression

initial distribution from ASTRA (200000 particles)

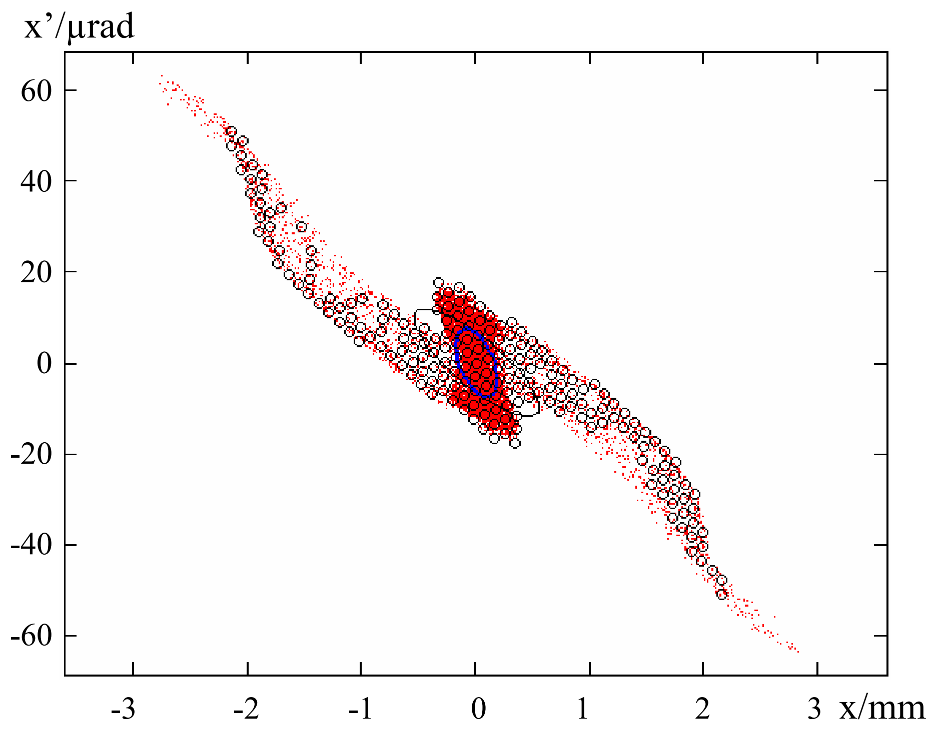
$$\gamma_0 = 266.6 \quad q_{tot} = 2.61\,\mathrm{nC}$$



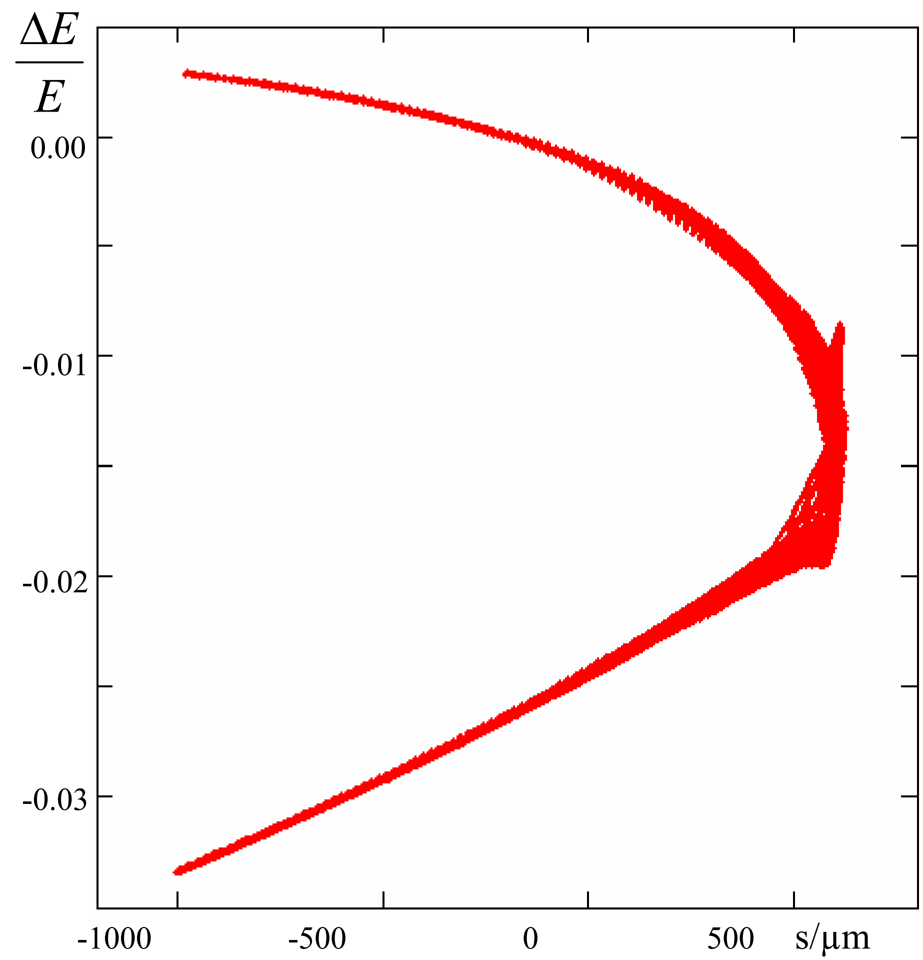
$\rightarrow$ 41000 sub-bunches (CSRtrack)

Example: BC2-TTF1

initial distribution
horizontal slice

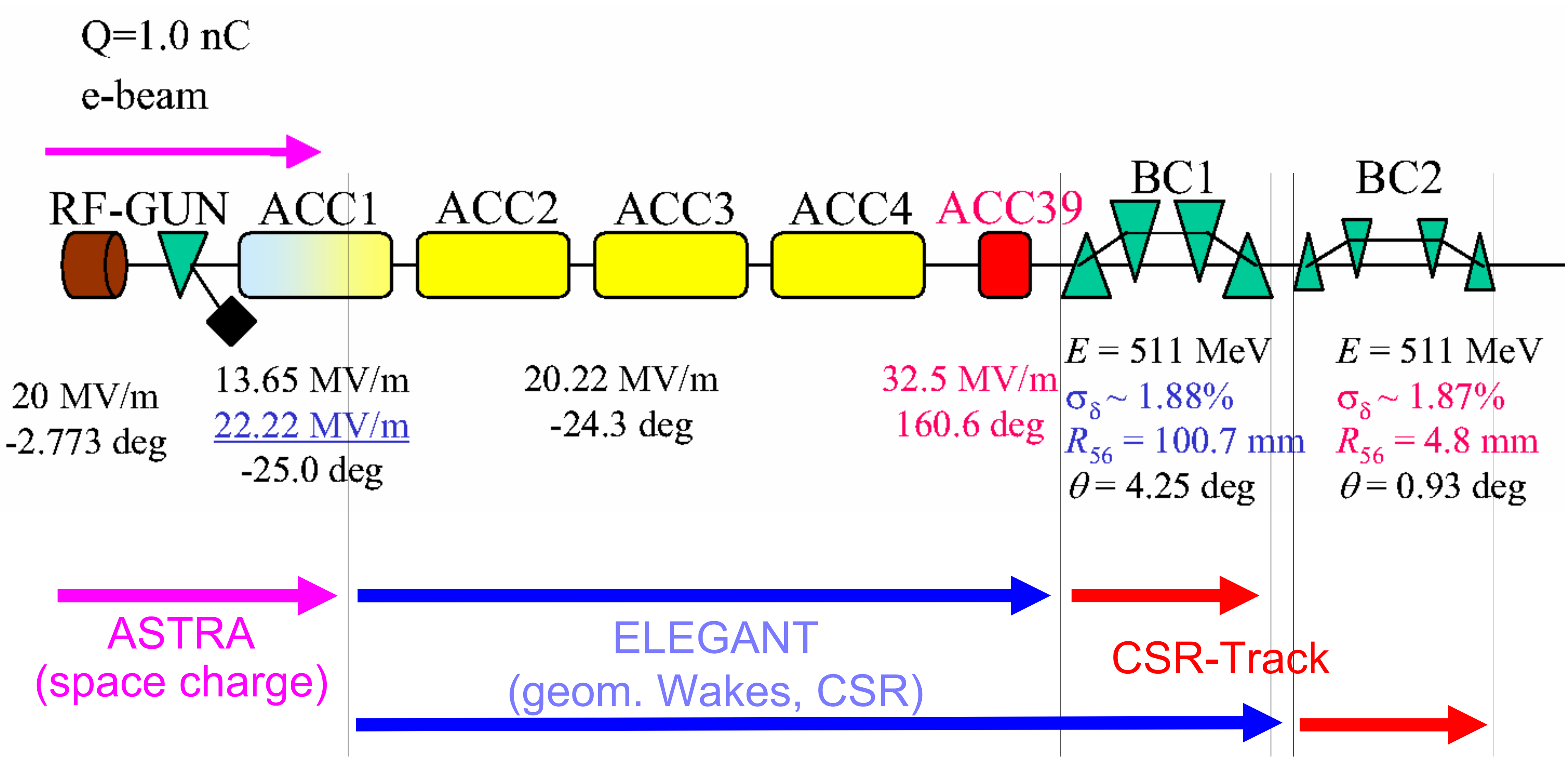


after BC:
longitudinal phase space



Example: double BC

(proposal for TTF2)



Example: double BC

p to p : 10d on 20x1GHz CPU
greens : 0.5d on 1x1GHz CPU

projected : ~15min on 1x1GHz CPU