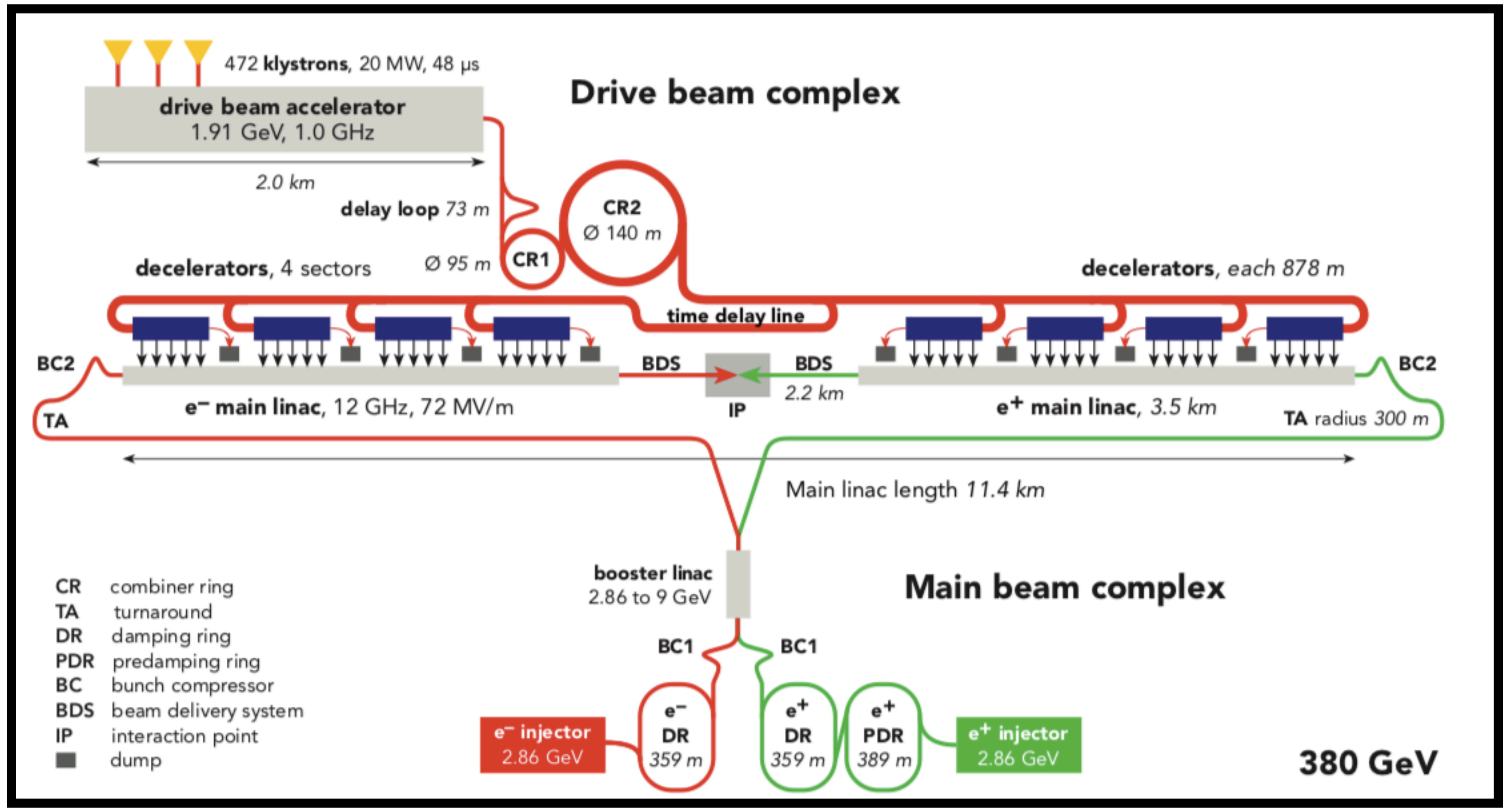
Tuning the CLIC 380 GeV final focus system

Two-beam tuning using realistic signals

Jim Ögren

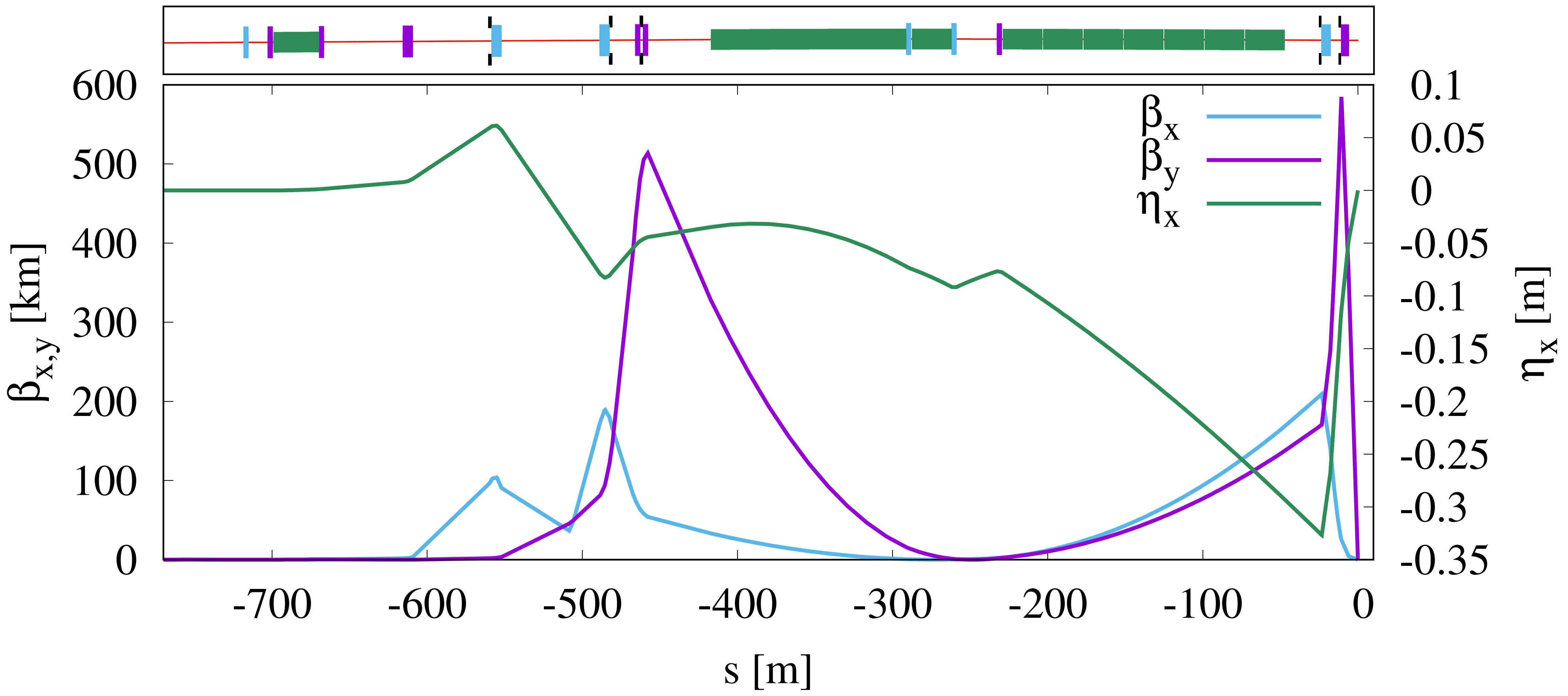
CLIC Beam physics meeting June 20, 2019

Recap: 380 GeV final-focus system



- Local chromaticity scheme
- 780 m long
- 20 quads, 6 sextupoles and 2 octupoles
- L* = 6 m

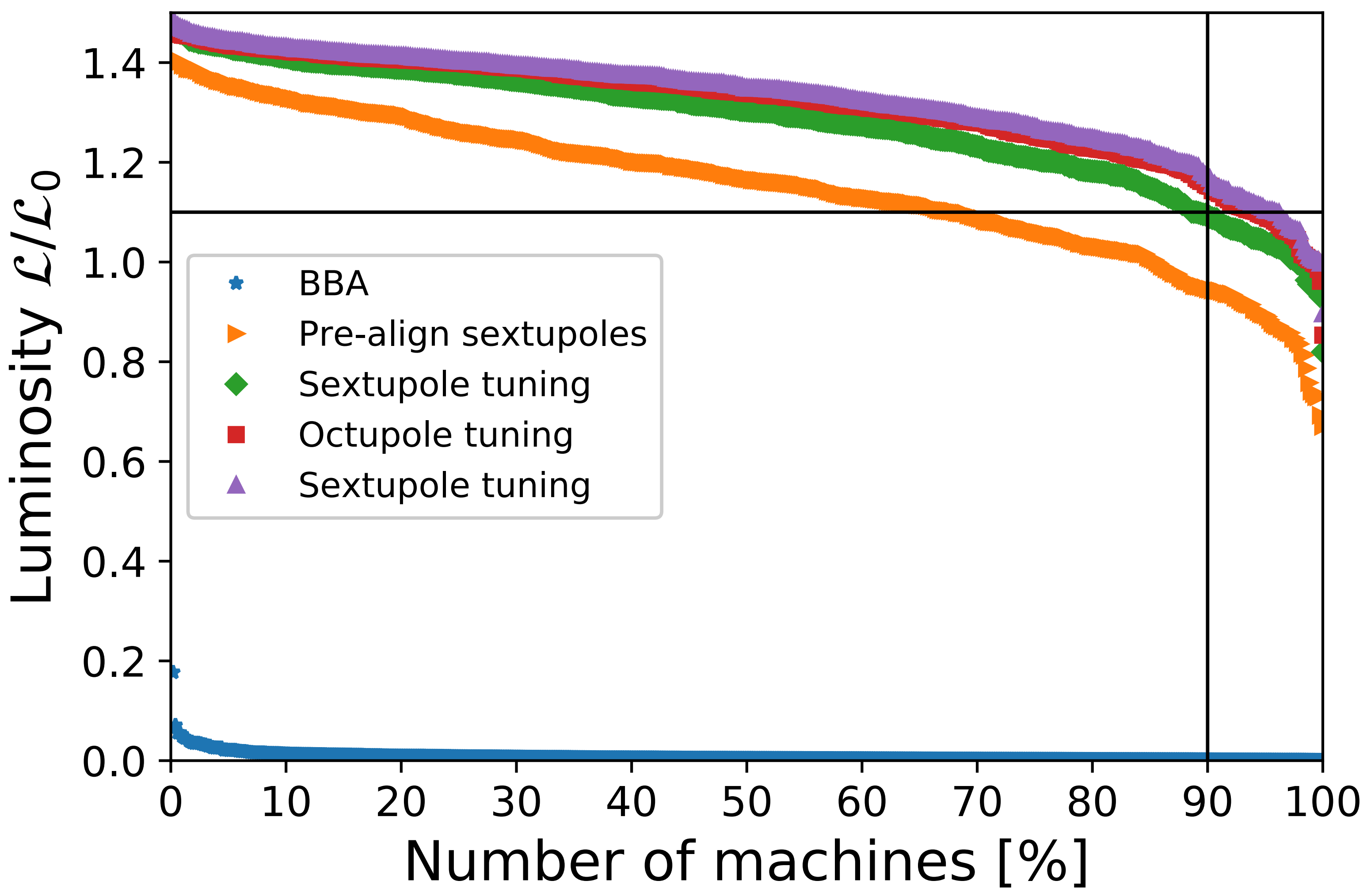
Norm. emittance (end of linac) $\gamma\epsilon_x/\gamma\epsilon_y$	[nm]	900 / 20
Norm. emittance (IP) $\gamma\epsilon_x/\gamma\epsilon_y$	[nm]	950 / 30
Beta function (IP) β_x^*/β_y^*	[mm]	8.2 / 0.1
Target IP beam size σ_x^*/σ_y^*	[nm]	149 / 2.9
Bunch length σ_z	[μm]	70
rms energy spread δ_p	[%]	0.35
Bunch population N_e	[10^9]	5.2
Number of bunches n_b		352
Repetition rate f_{rep}	[Hz]	50
Luminosity $\mathcal{L}_{total}$	[10^{34}cm^{-2}s^{-1}]	1.5
Peak luminosity $\mathcal{L}_{1\%}$	[10^{34}cm^{-2}s^{-1}]	0.9

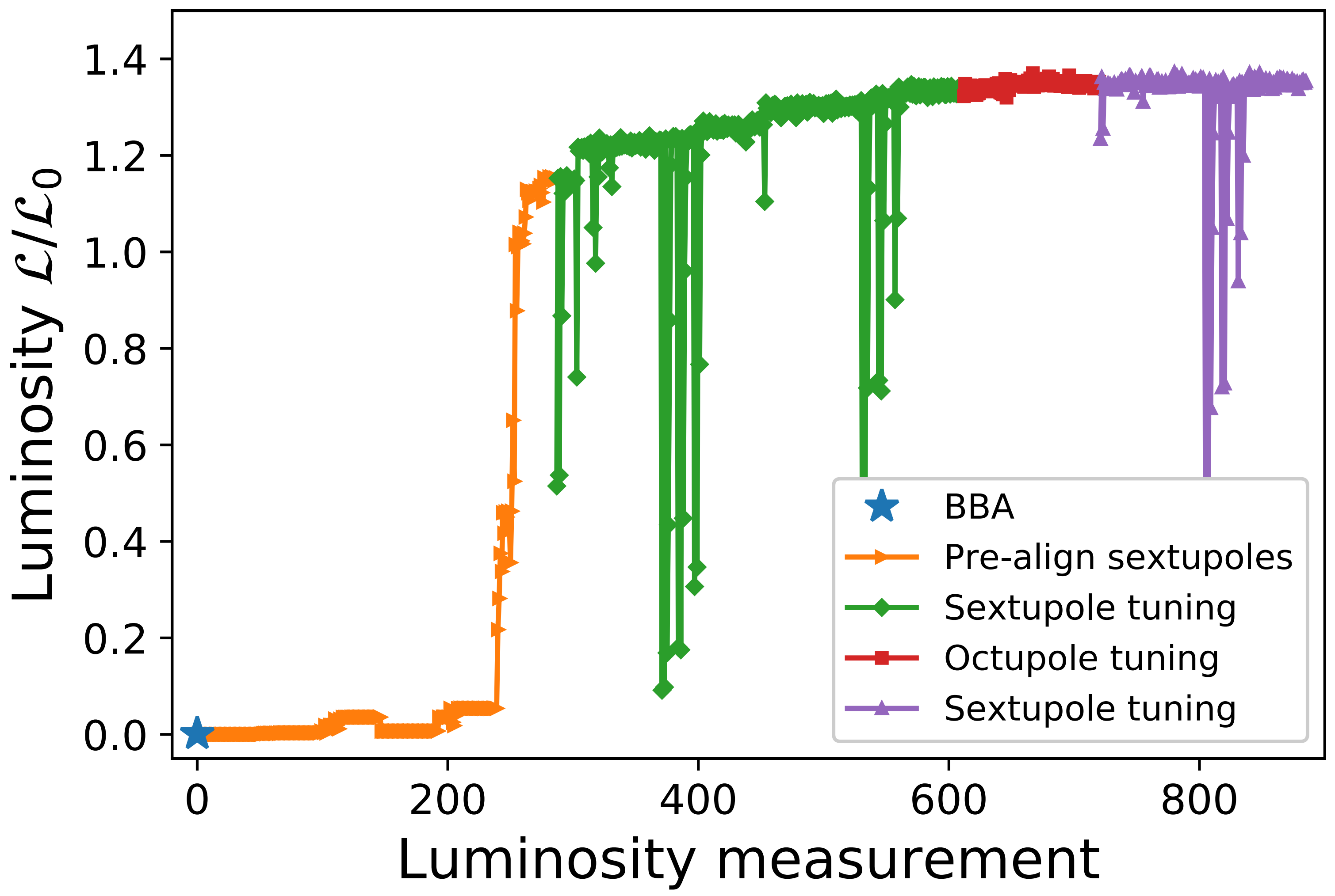


Recap: One-beam tuning

500 machines, randomly distributed imperfections

- Left: 95% of machines reach goal of 110% of L_0
- Right: Luminosity evolution for the median machine



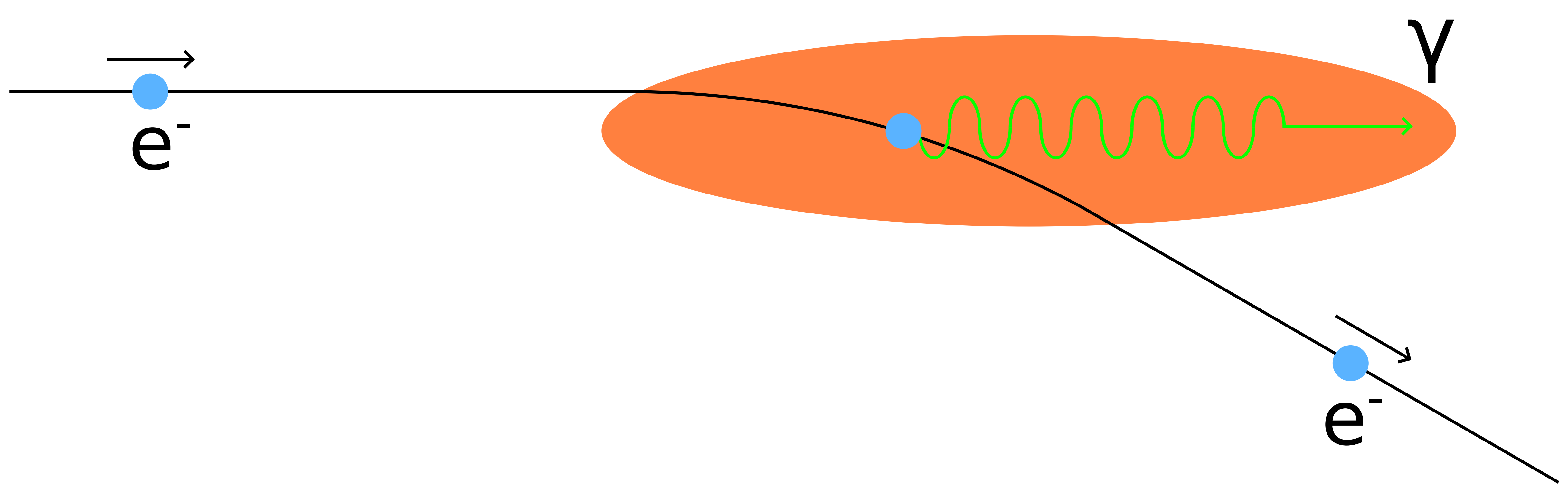


- 95% reached goal in 900 luminosity measurements

- **Today:** two-beam tuning, static imperfections
  - Independent imperfections for e^-/e^+ beamlines
  - Twice the computations needed for tracking

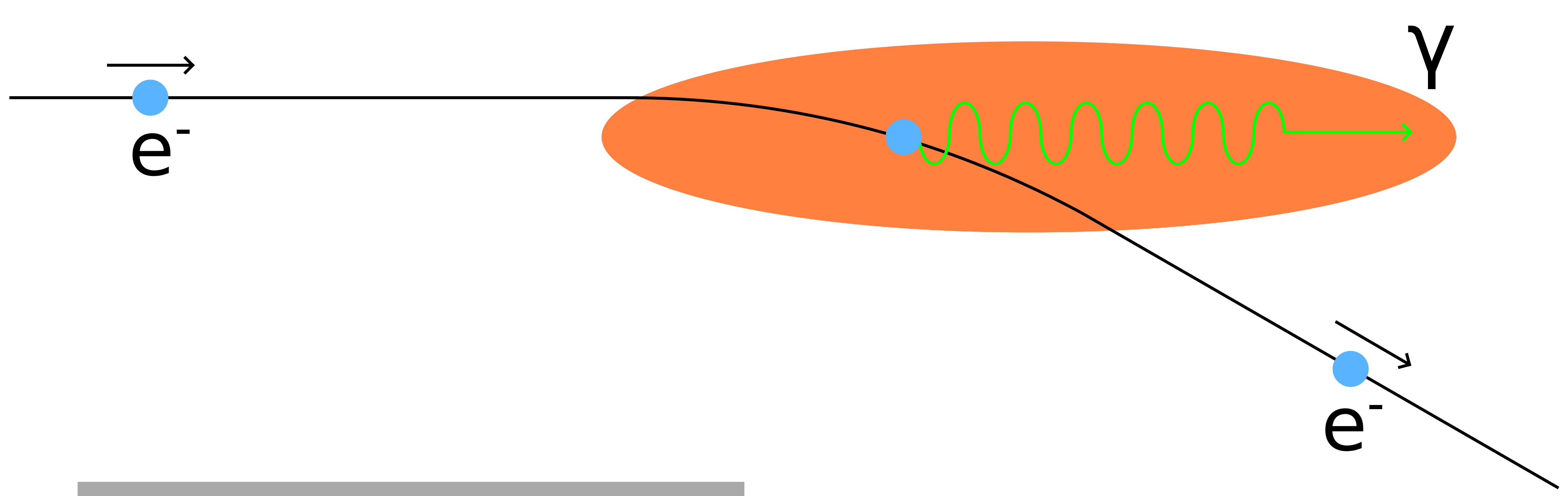
- Using realistic signals (i.e. assume luminosity is difficult/time-consuming to measure)

Beamstrahlung



Beamstrahlung:
Particles in the one beam are bent by the EM fields from the other beam during collision. This leads to synchrotron radiation.

Beamstrahlung



Beamstrahlung:
Particles in the one beam are bent by the EM fields from the other beam during collision. This leads to synchrotron radiation.

$$n_\gamma \propto \frac{N}{\sigma_x + \sigma_y}$$

$$E_\gamma \propto \frac{N}{(\sigma_x + \sigma_y)\sigma_z}$$

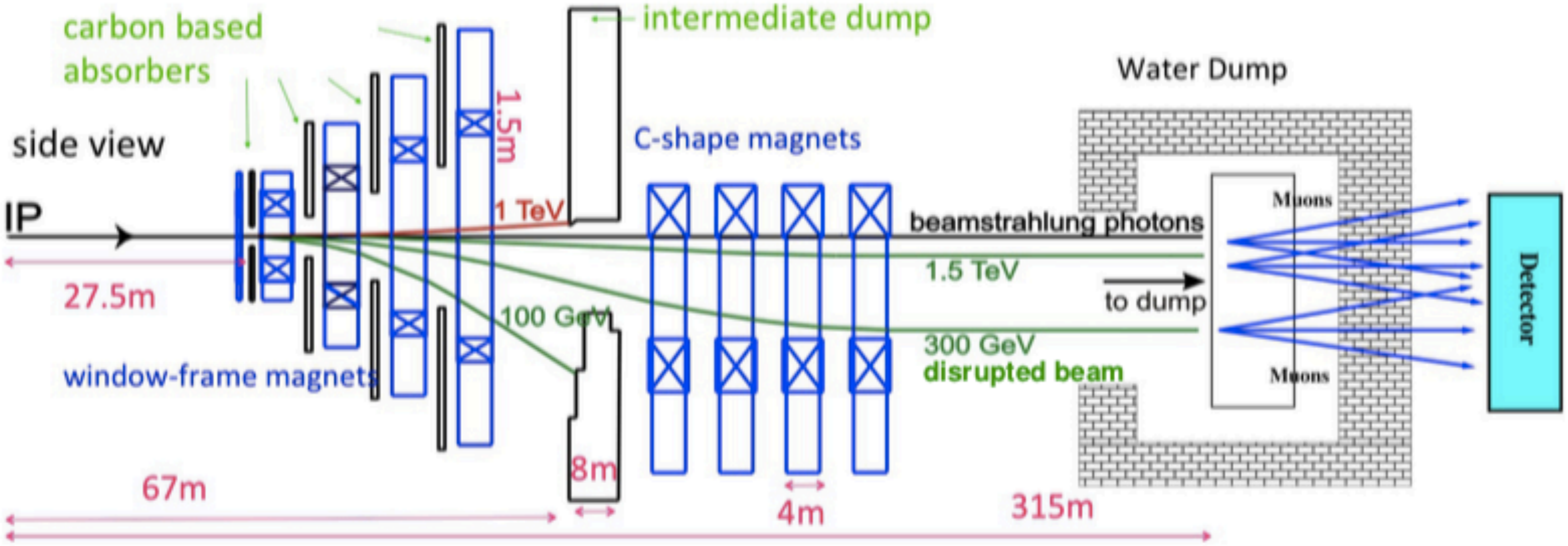
$$\mathscr{L} \propto \frac{N^2}{\sigma_x \sigma_y}$$

Number of photons and photon energy is proportional to $1/(\sigma_x + \sigma_y)$ and Luminosity $1/\sigma_x\sigma_y$

=> use flat beams!

Beamstrahlung can also be used as a beam size indicator

Measuring beamstrahlung

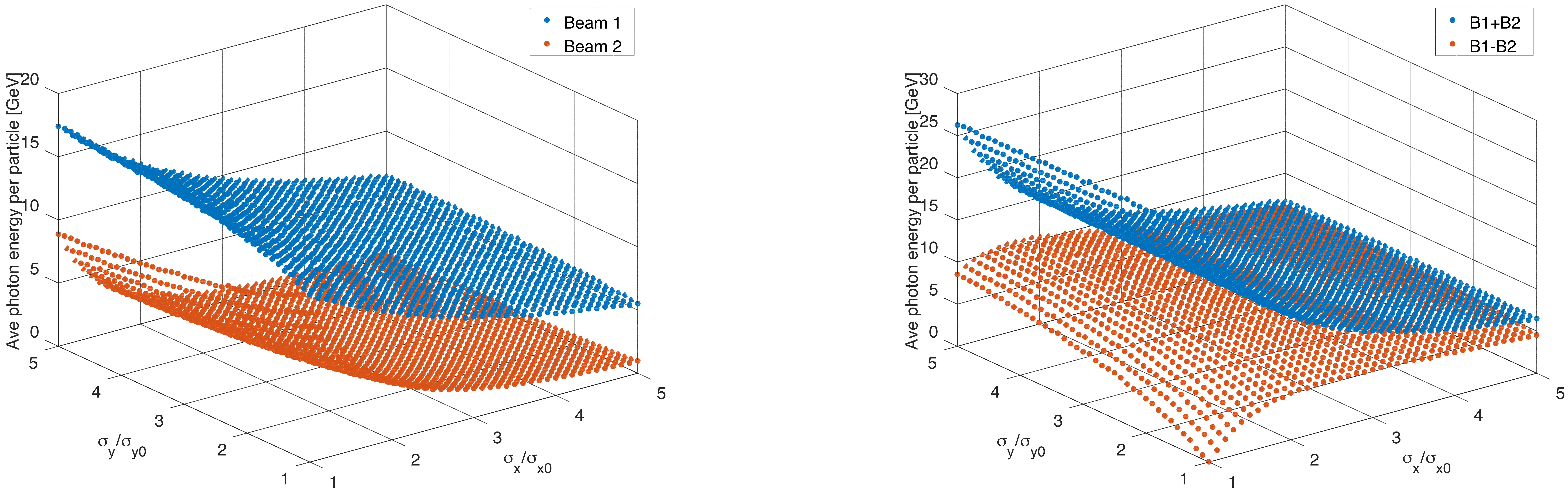


A. Apyan et al., *CLIC Luminosity Monitoring*, In Proc., IPAC'12.

Beamstrahlung can be measured using Cherenkov detectors measuring muons created from photon interaction in the water dump

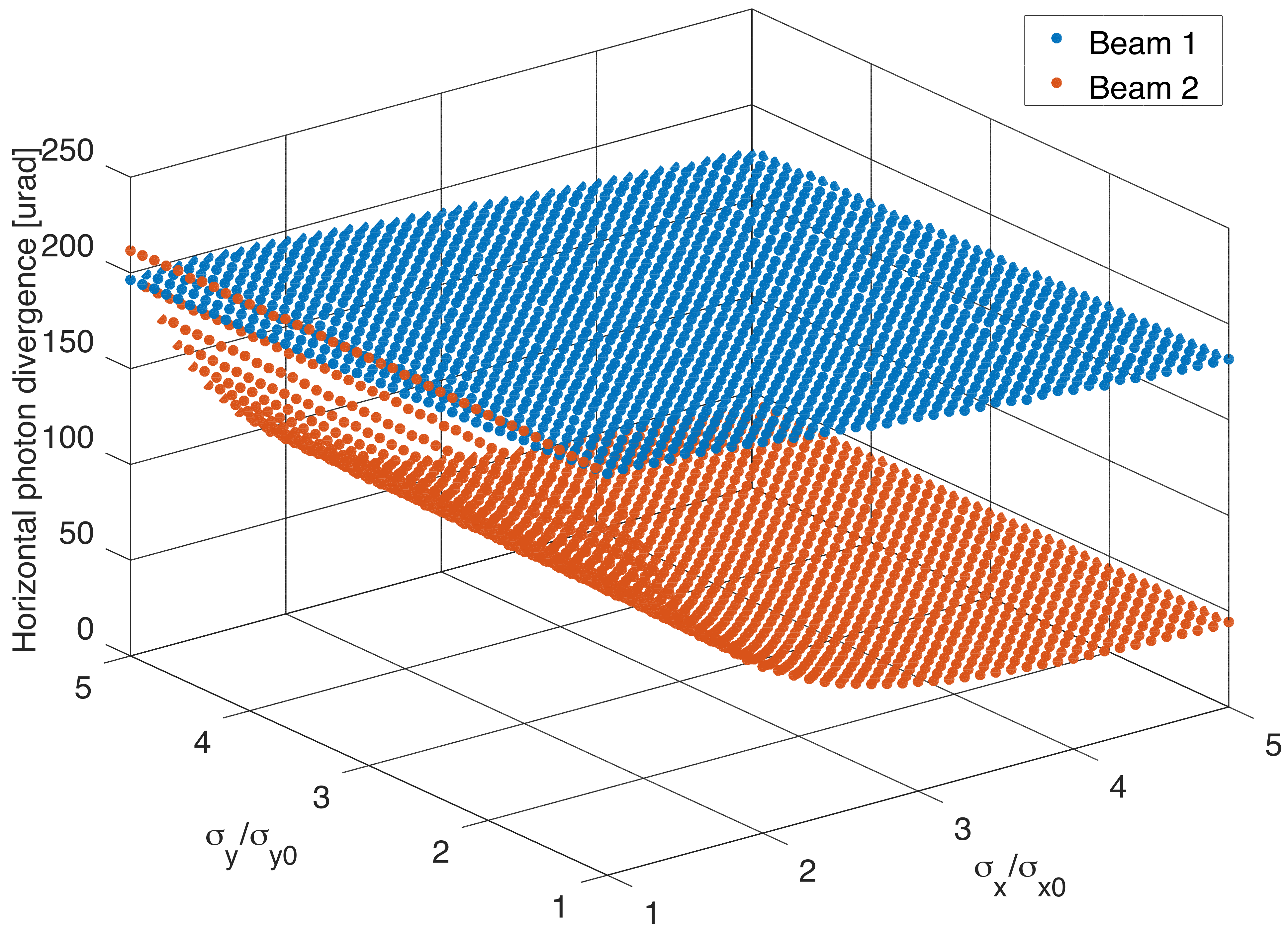
The total beamstrahlung power is in the order of hundreds of kW

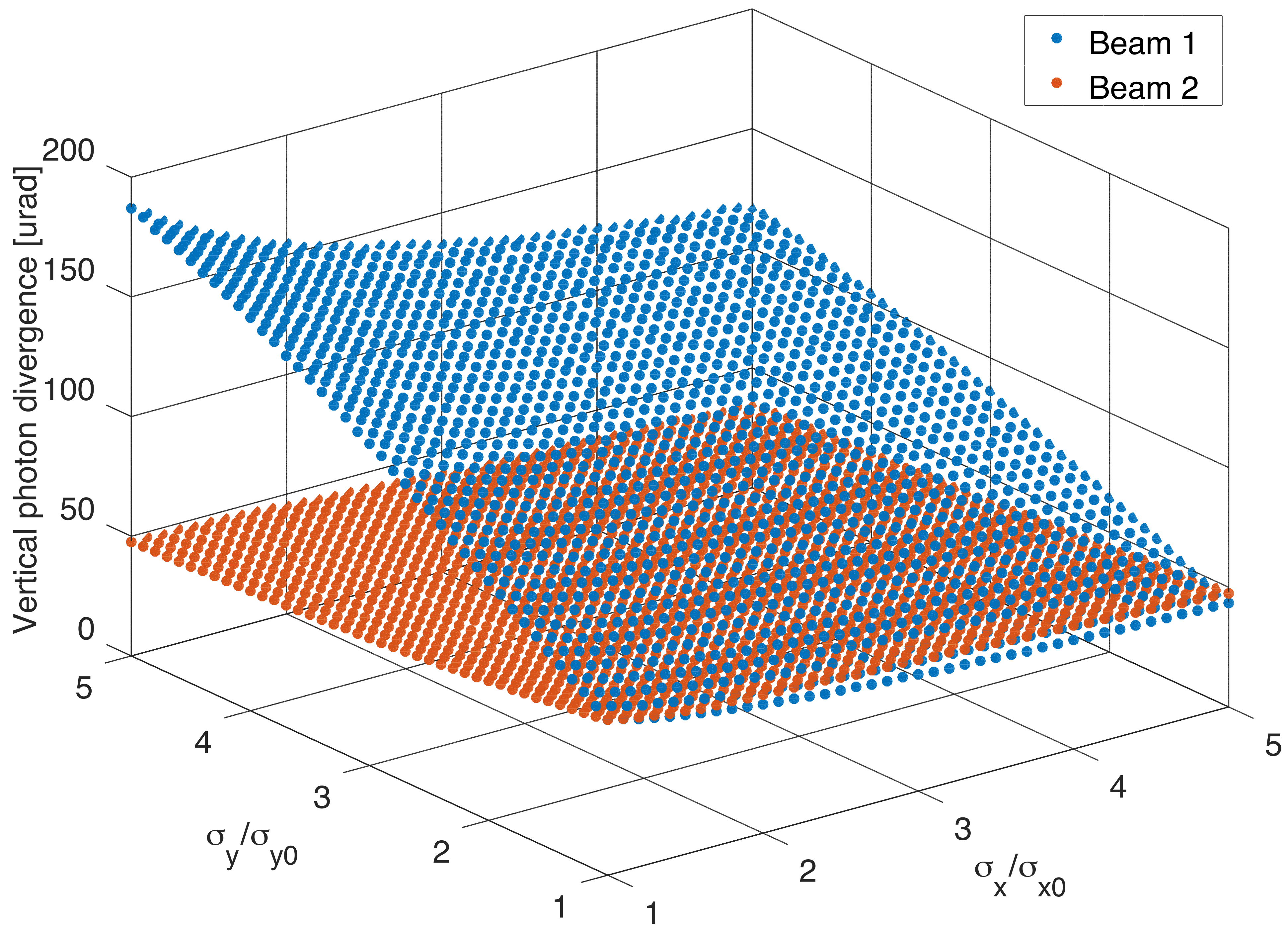
Tuning signals: beamstrahlung



- Strong dependence on horizontal beam size
- First order: total power as horizontal beam size signal
- From beamstrahlung asymmetry we can determine which beam is larger
- How to determine vertical beam size?

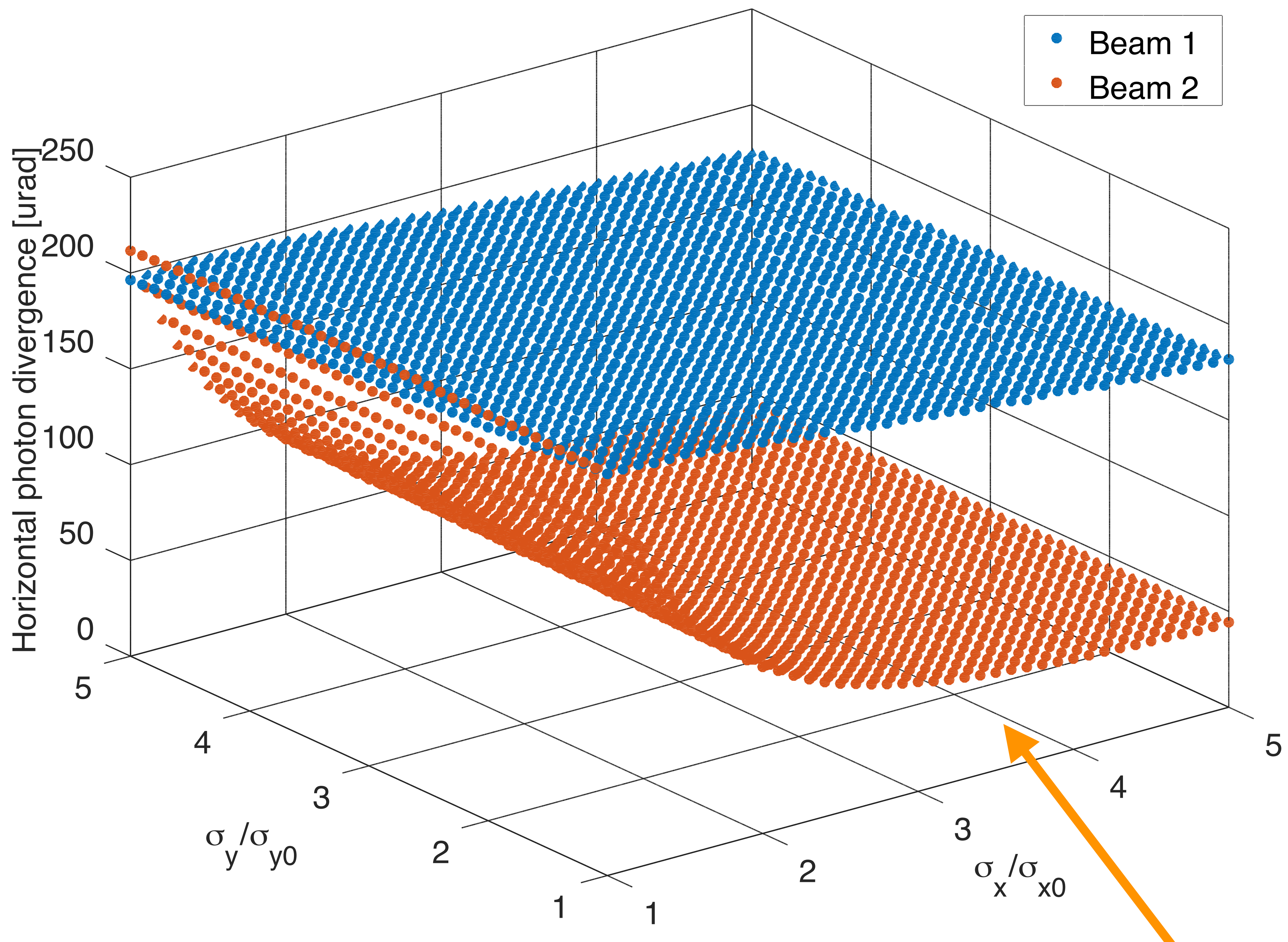
Beamstrahlung: photon divergence

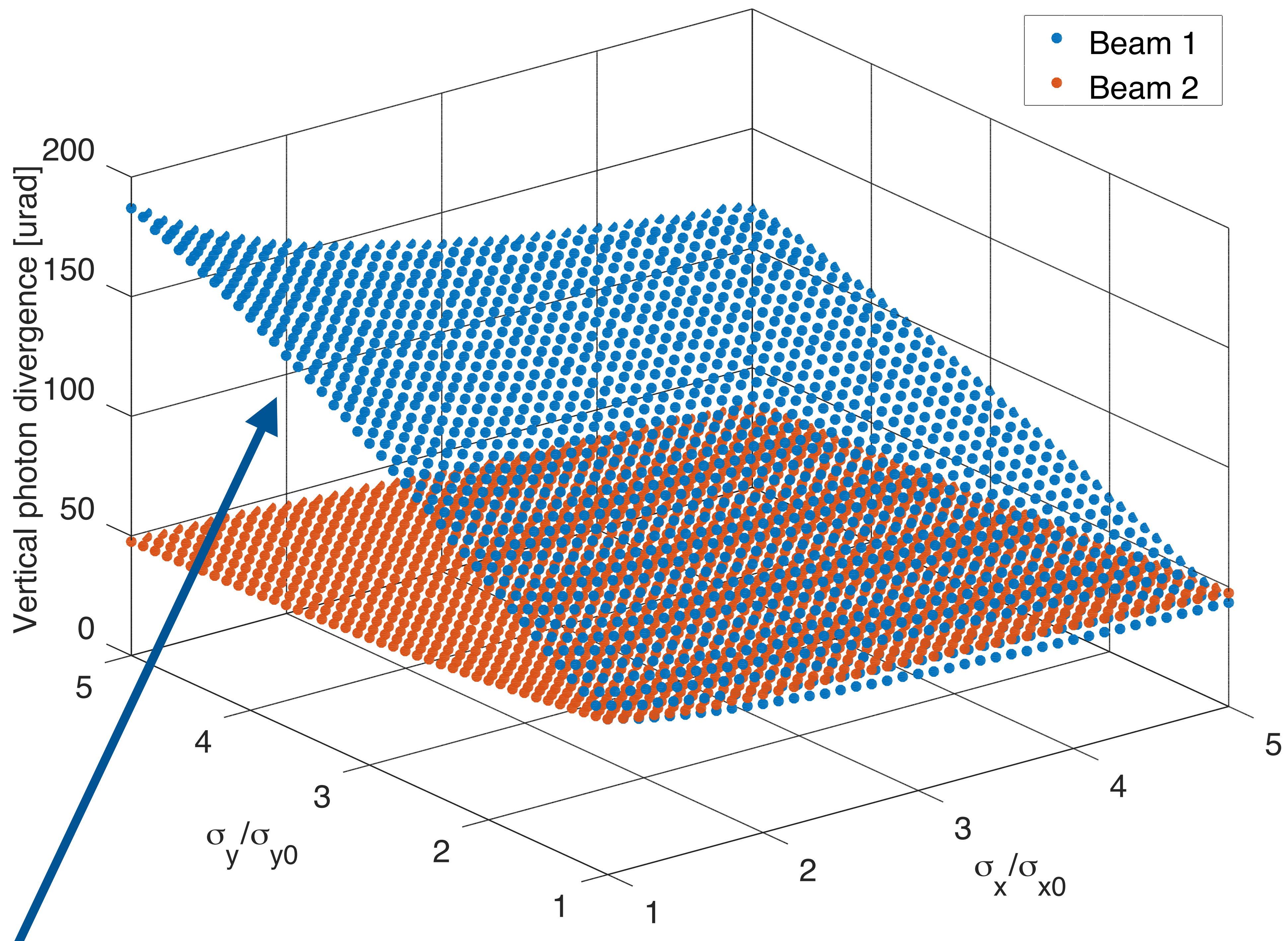




- Vertical divergence of photon beam is a good estimator of vertical beam size
- How to distinguish the effect from increasing horizontal and vertical beam size?

Beamstrahlung: photon divergence

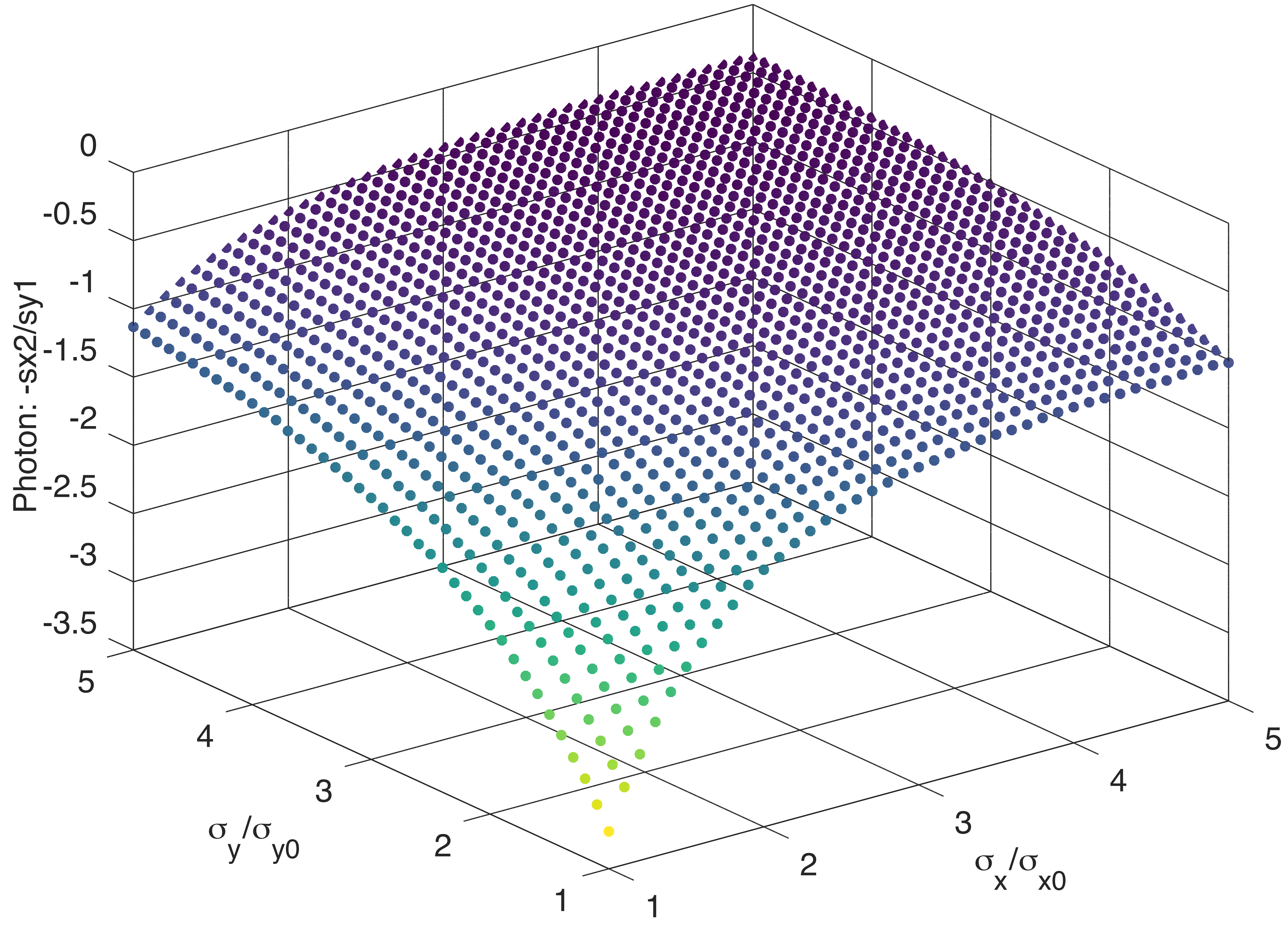




- Vertical divergence of photon beam is a good estimator of vertical beam size
- How to distinguish the effect from increasing horizontal and vertical beam size?

Use a combined signal: $Y = \sigma_{x'}^{(2)}/\sigma_{y'}^{(1)}$

Signal for vertical beam size?



- Combined signal:

$$Y = -\sigma_{x'}^{(2)}/\sigma_{y'}^{(1)}$$

- Signal for both horizontal and vertical beam size
- Beamstrahlung for beam size tuning
- Need signal for Luminosity

Incoherent pairs

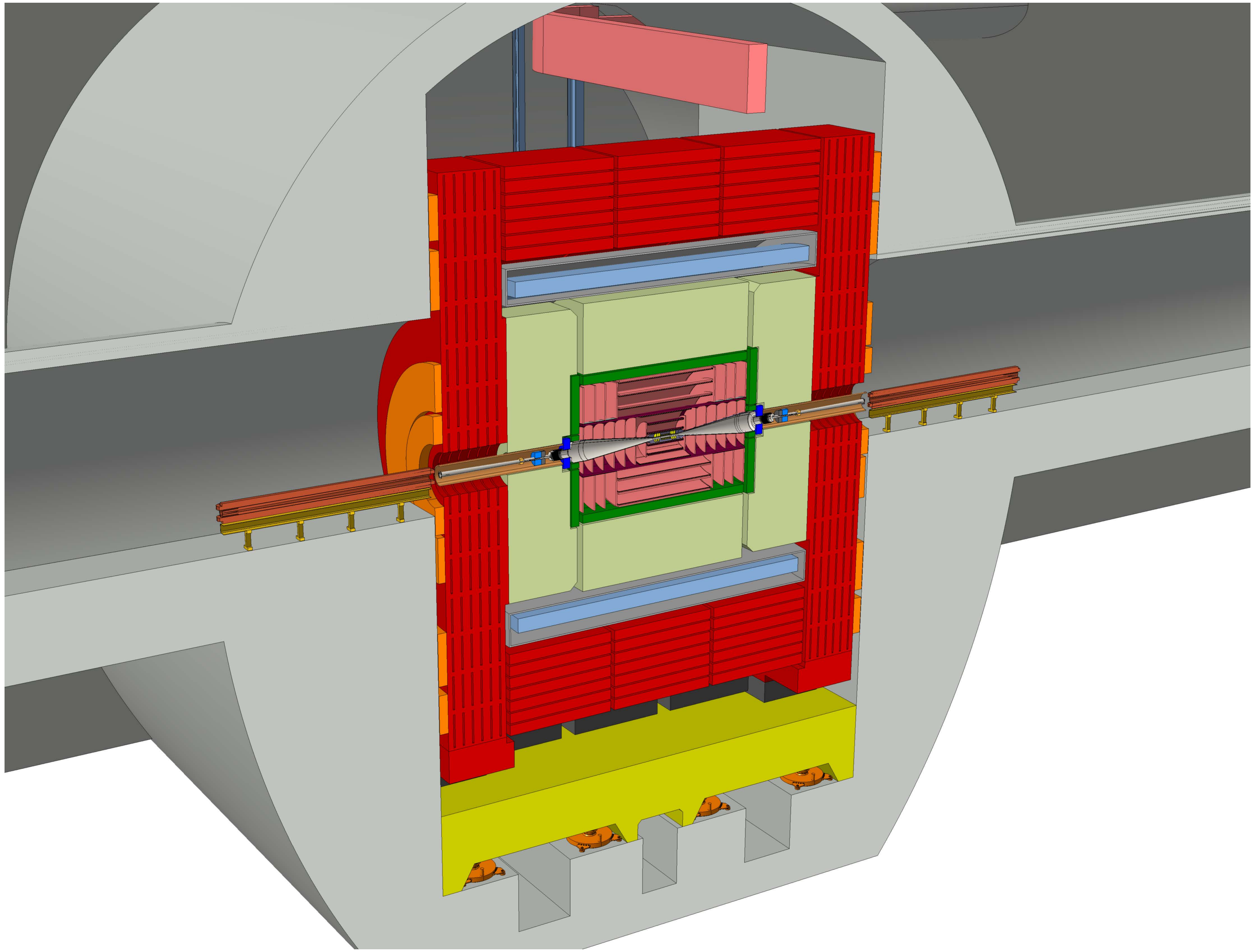
Three mechanisms for incoherent e^-e^+ production:

- **Breit—Wheeler:** $\gamma + \gamma \rightarrow e^- + e^+$ two real photons
- **Bethe—Heitler:** $\gamma + e^\pm \rightarrow e^\pm + e^- + e^+$ one real and one virtual photon
- **Landau—Lifshitz:** $e + e \rightarrow e + e + e^- + e^+$ two virtual photons

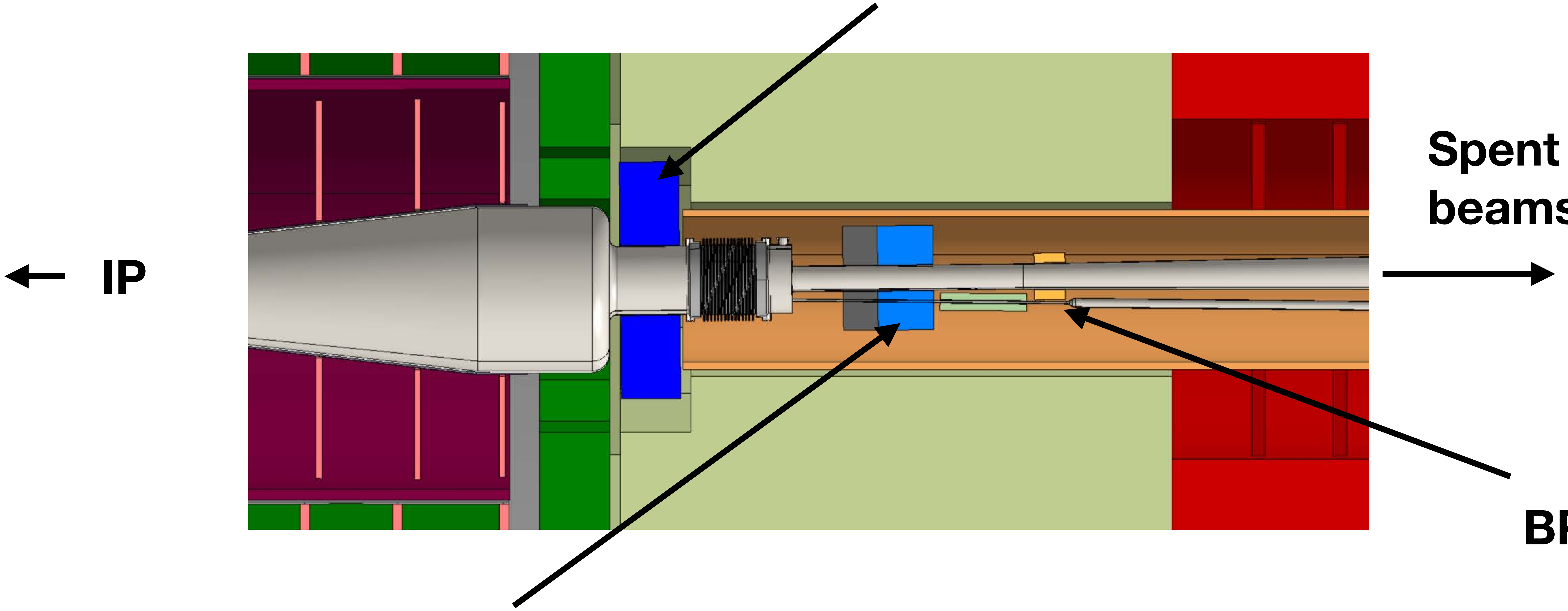
There is also **coherent pair** production where a beamstrahlung photon interact with the macroscopic field of the opposing beam

- Requires very strong fields
- **Negligible for CLIC 380 GeV** (but not for CLIC 3 TeV)

The CLIC Detector



The BeamCal



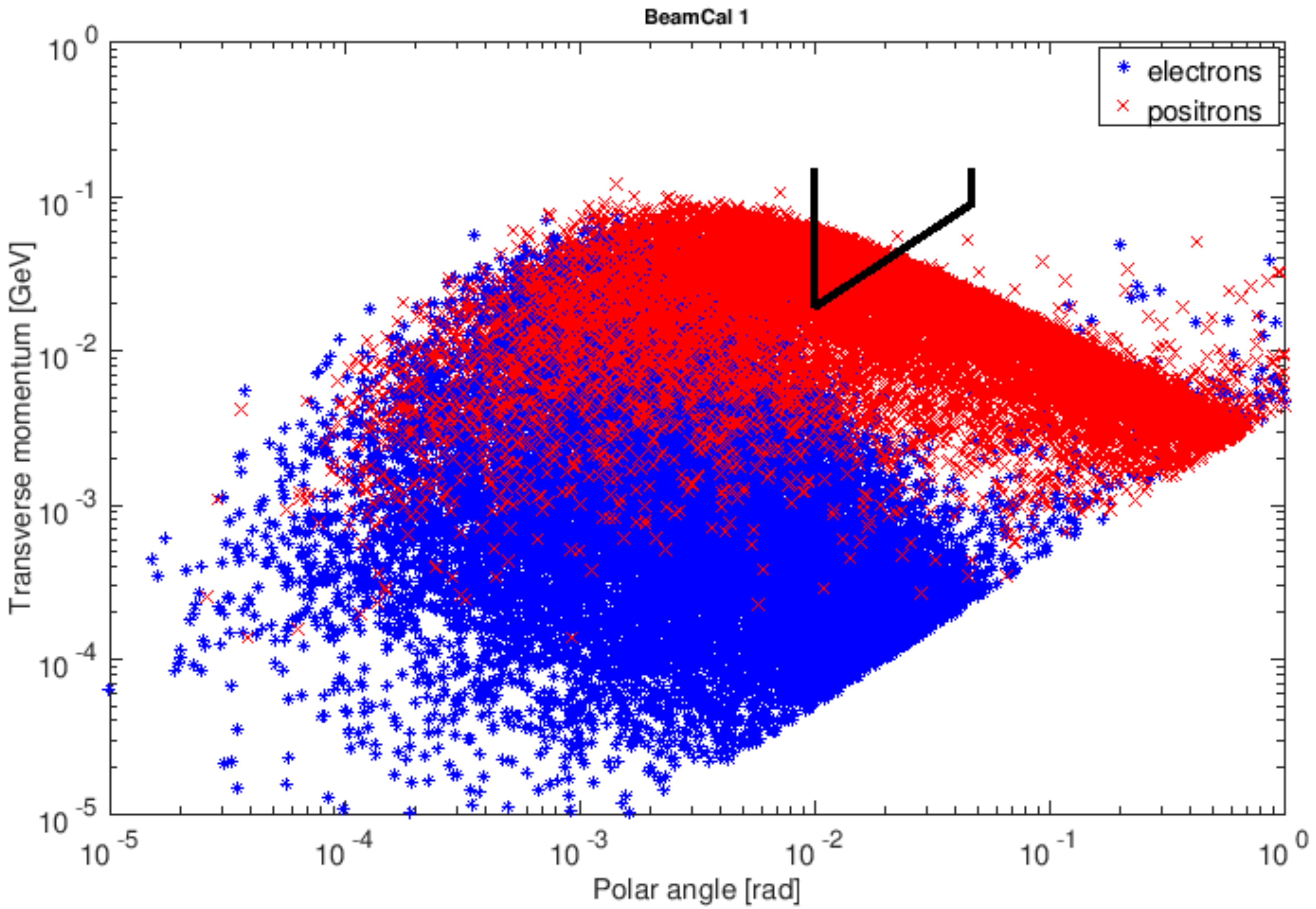
	Z_{start} [mm]	Z_{end} [mm]	R_{in} [mm]	R_{out} [mm]	θ_{min} [mrad]	θ_{max} [mrad]
LumiCal	2539	2710	100	340	39	134
BeamCal	3181	3441	32	150	10	46

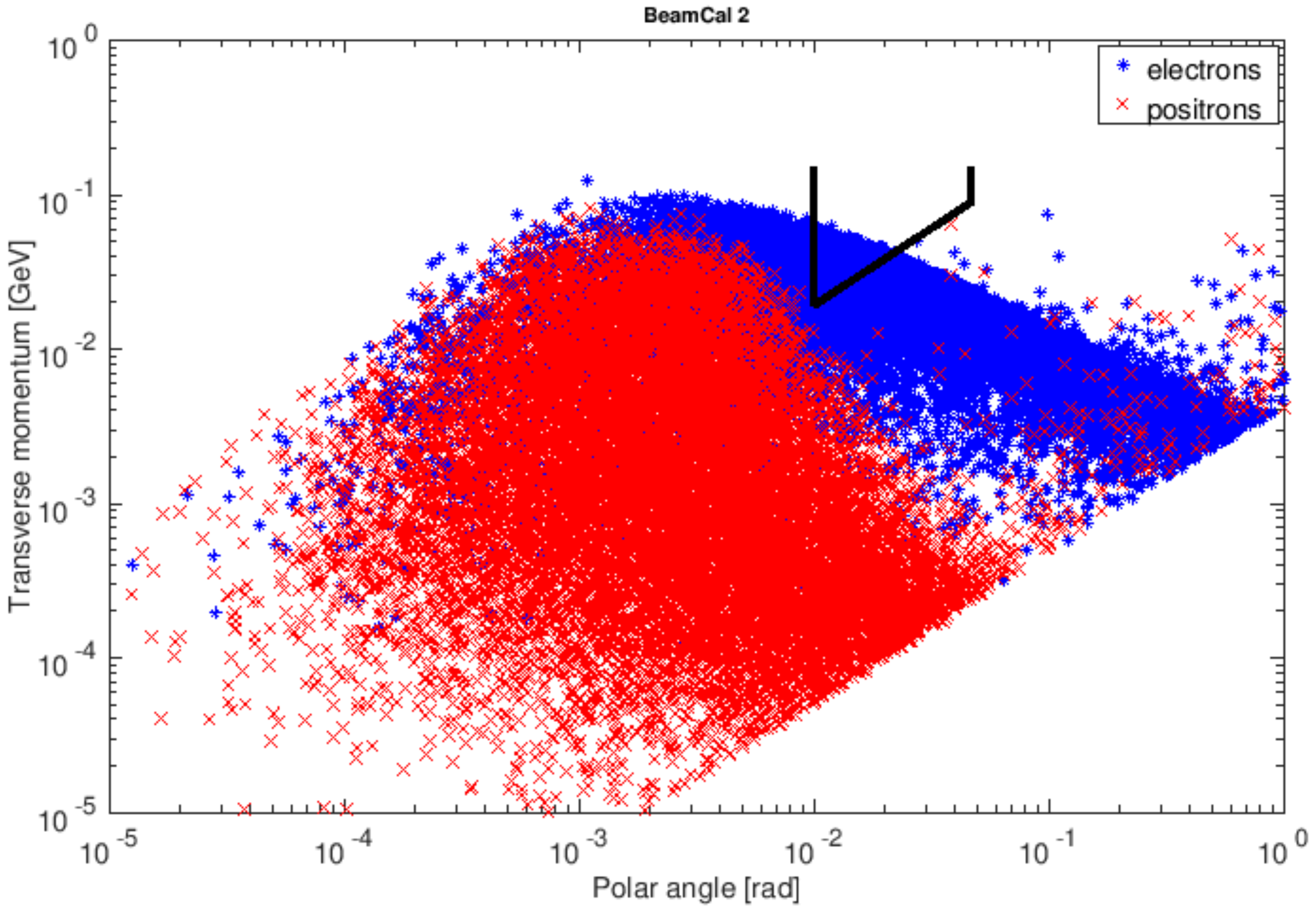
Incoherent pairs in BeamCal

- Compute and track pairs in GUINEA-PIG
  - deflection from EM fields of beam, use 7 grids
- Make cut in p_T—θ plane
  - only particles with correct angles and high enough energy
  - Total energy of ~6000 GeV/bunch-crossing is deposited in the two BeamCals
- Use incoherent pairs as quick luminosity estimator?

Incoherent pairs in BeamCal

- Compute and track pairs in GUINEA-PIG
  - deflection from EM fields of beam, use 7 grids
- Make cut in p_T—θ plane
  - only particles with correct angles and high enough energy
  - Total energy of ~6000 GeV/bunch-crossing is deposited in the two BeamCals
- Use incoherent pairs as quick luminosity estimator?

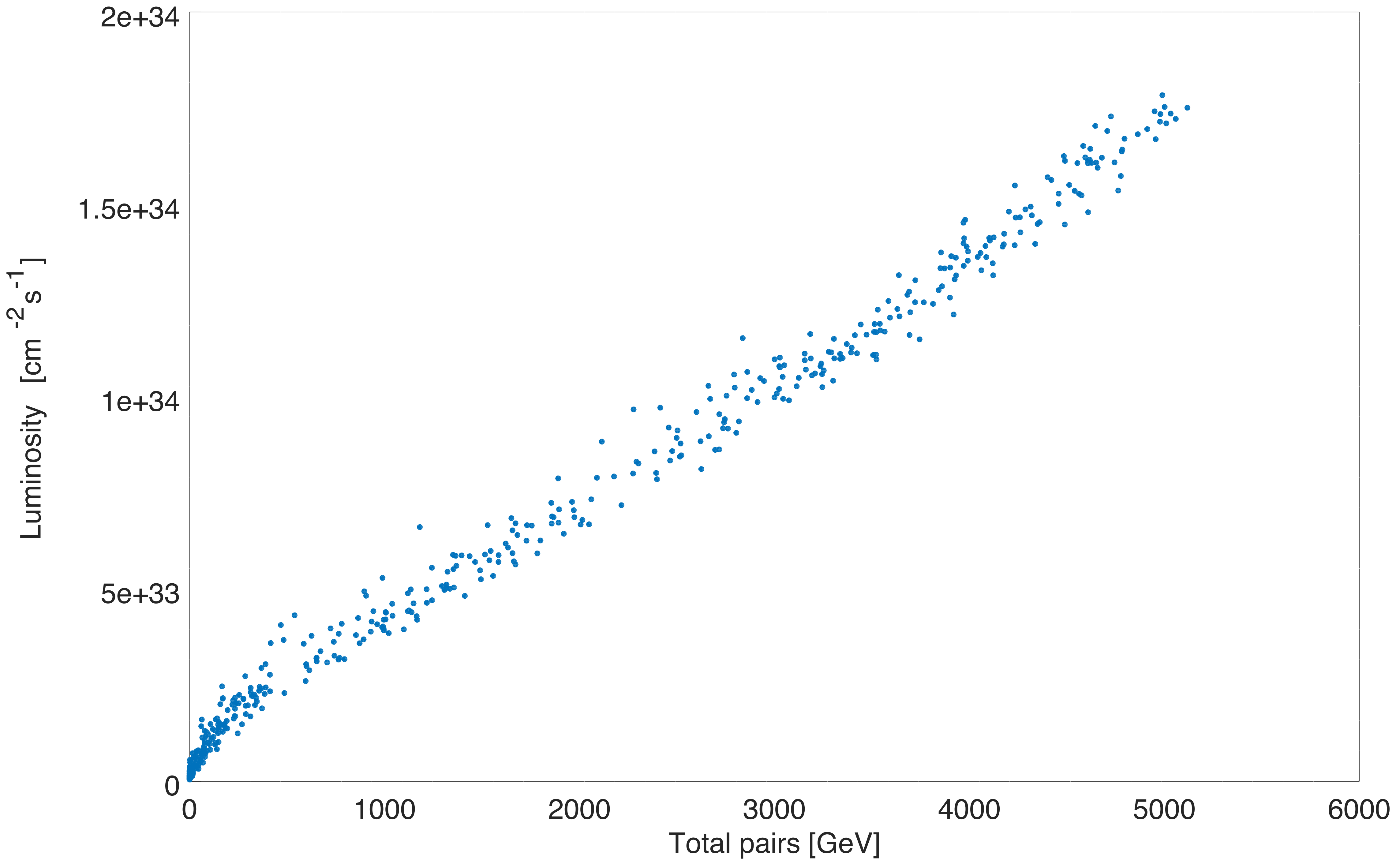




Tuning signals

Perfect machine with sext. offsets

- Generated 1000 machines with sextupole random misalignments
- Linear correlation between luminosity and pairs
- Threshold value
- However, quite noisy signal



Simulation: tuning with realistic signals

Static imperfections:

Imperfection	Specified tolerance (rms error)	Elements
Resolution	20 nm	BPMs
Transverse misalignments	10 µm	BPMs, quadrupoles, multipoles
Roll errors	100 µrad	BPMs, quadrupoles, multipoles
Relative strength error	10^{-4}	Quadrupoles, multipoles

In this study:

- BBA machines from single-beam study
- Beam from integrated simulation
- Seed of 150 machines

Monte Carlo simulations:

- Generate machines with random imperfections
- Luminosity goal: 110% of $L_0 = 1.5e34\ cm^{-2}s^{-1}$
- Tuning goal: 90% of machines to be successfully tuned

Simulation: tuning with realistic signals

- **Beam-based alignment**
    - Independent beamlines
    - Trajectory and dispersion

- **Beamstrahlung 1**
    - Maximize total beamstrahlung power
    - Tune larger beam, sextupole transverse position, random walk
    - Tunes mainly horizontal beam size

- **Beamstrahlung 2**
    - Tune vertical beam size, signal: combined signal with photon divergences and total power
    - Tune larger beam, sextupole transverse position, random walk

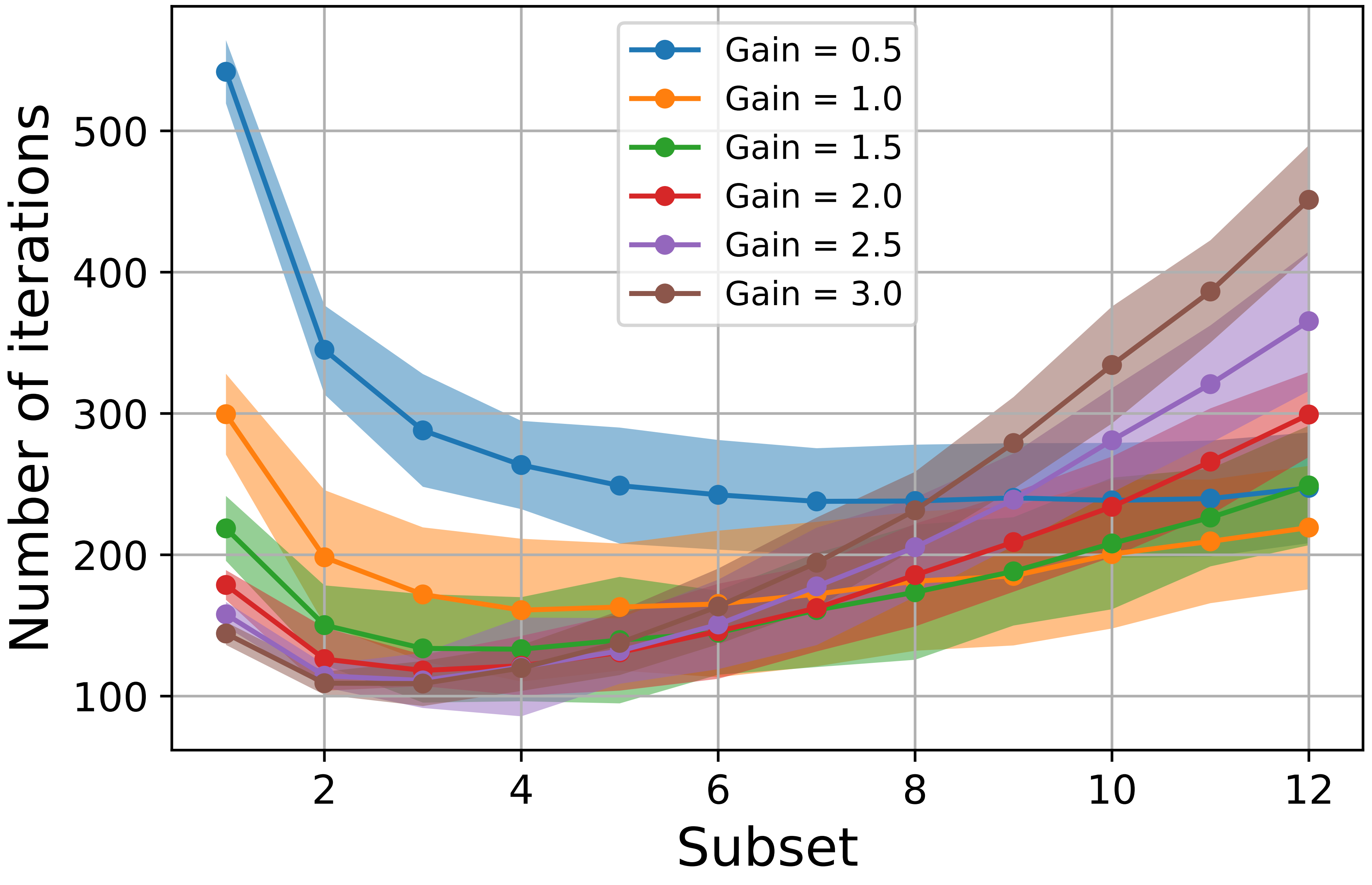
- **Sextupole knobs**
    - Scan sextupole knobs (transverse position)
    - Maximize energy deposited from inc. pairs in BeamCal
    - Use 2e4 particles and then 1e5 particles for fine tuning

Random walk

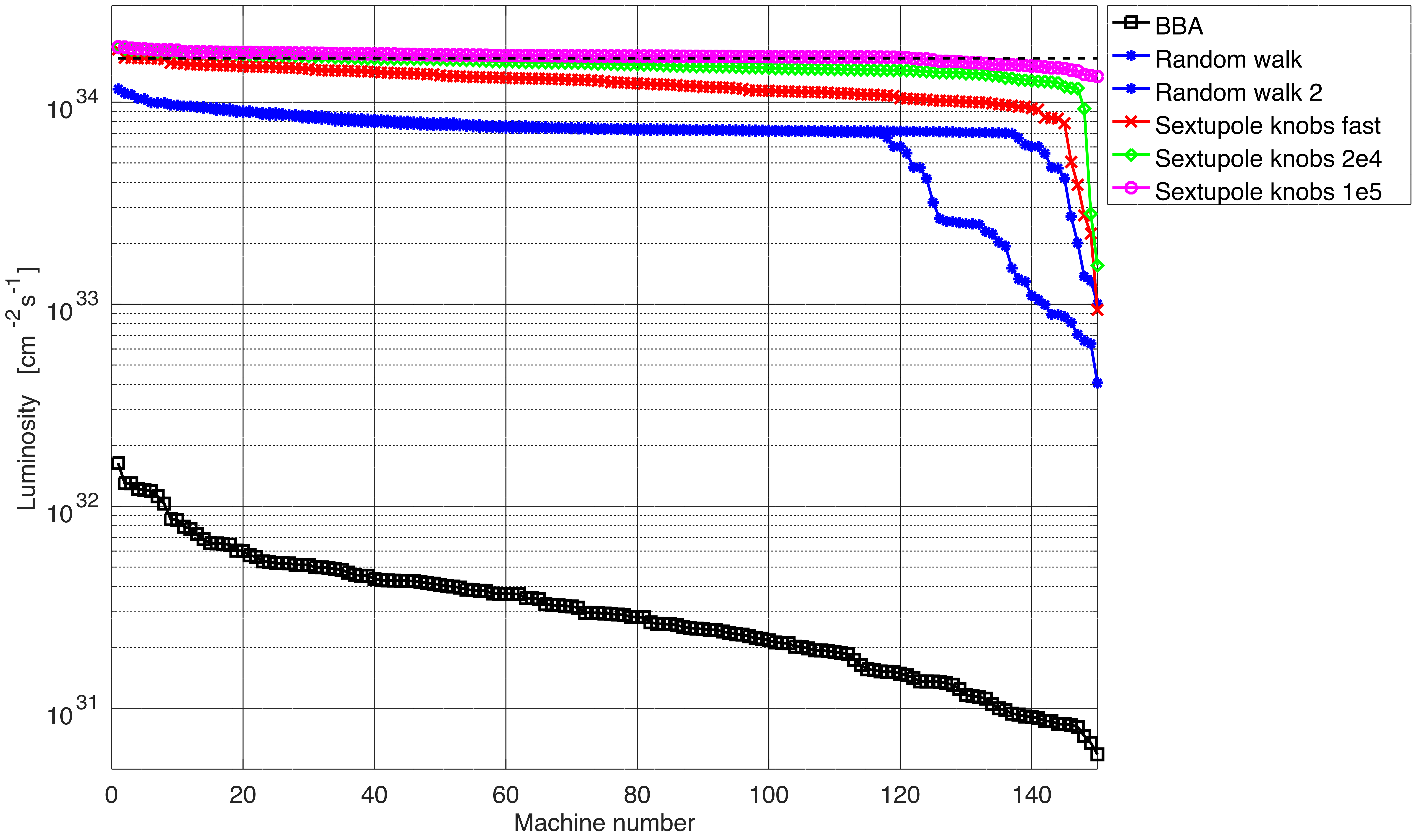
Random walk algorithm for sextupole transverse position

- Select a random subset (e.g. 6 out of 12 DOF)
- Select a random direction for that subset
- Perform a short (5 iterations) golden search
- Select point that optimizes signal
- Iterate

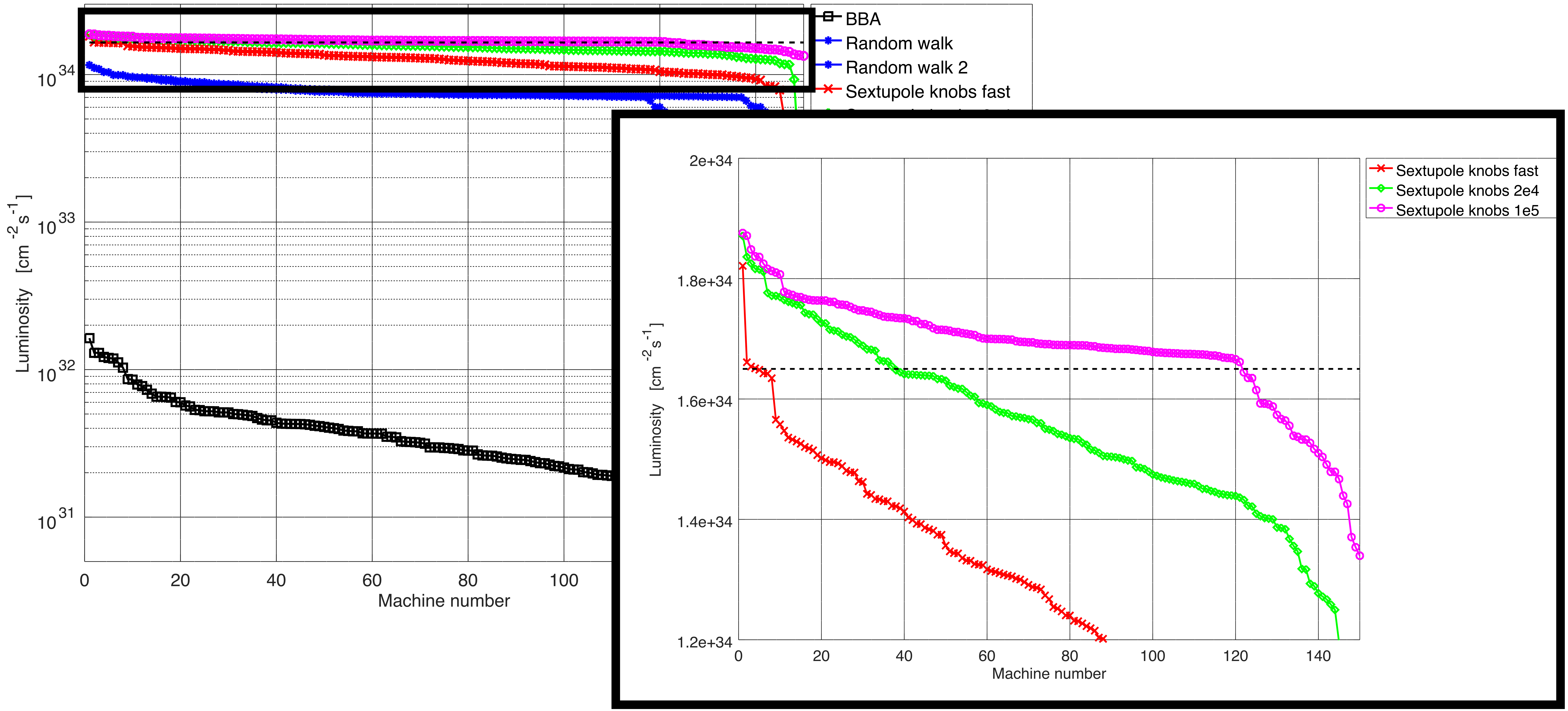
From optimization using NNs



Results: Luminosity histogram



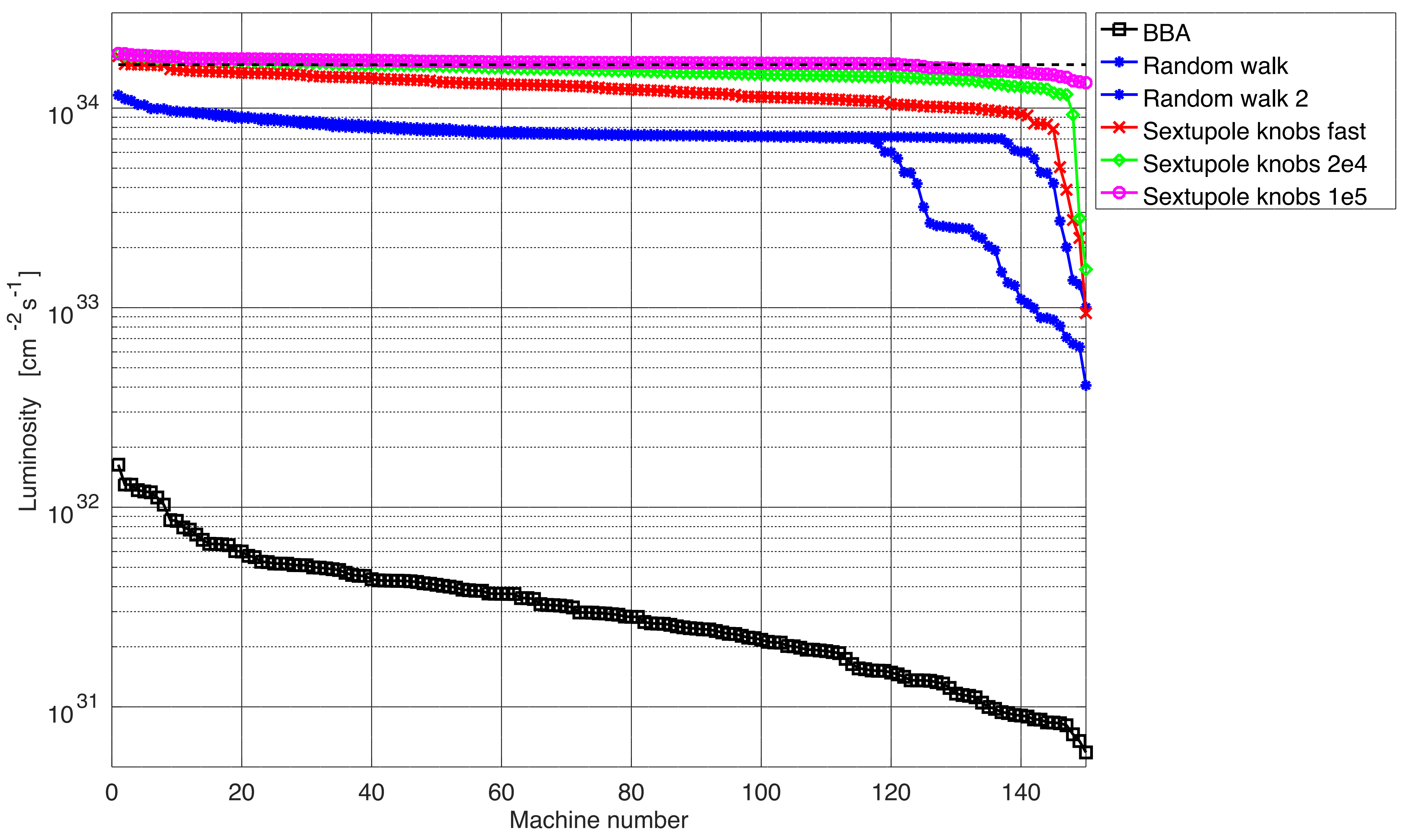
Results: Luminosity histogram



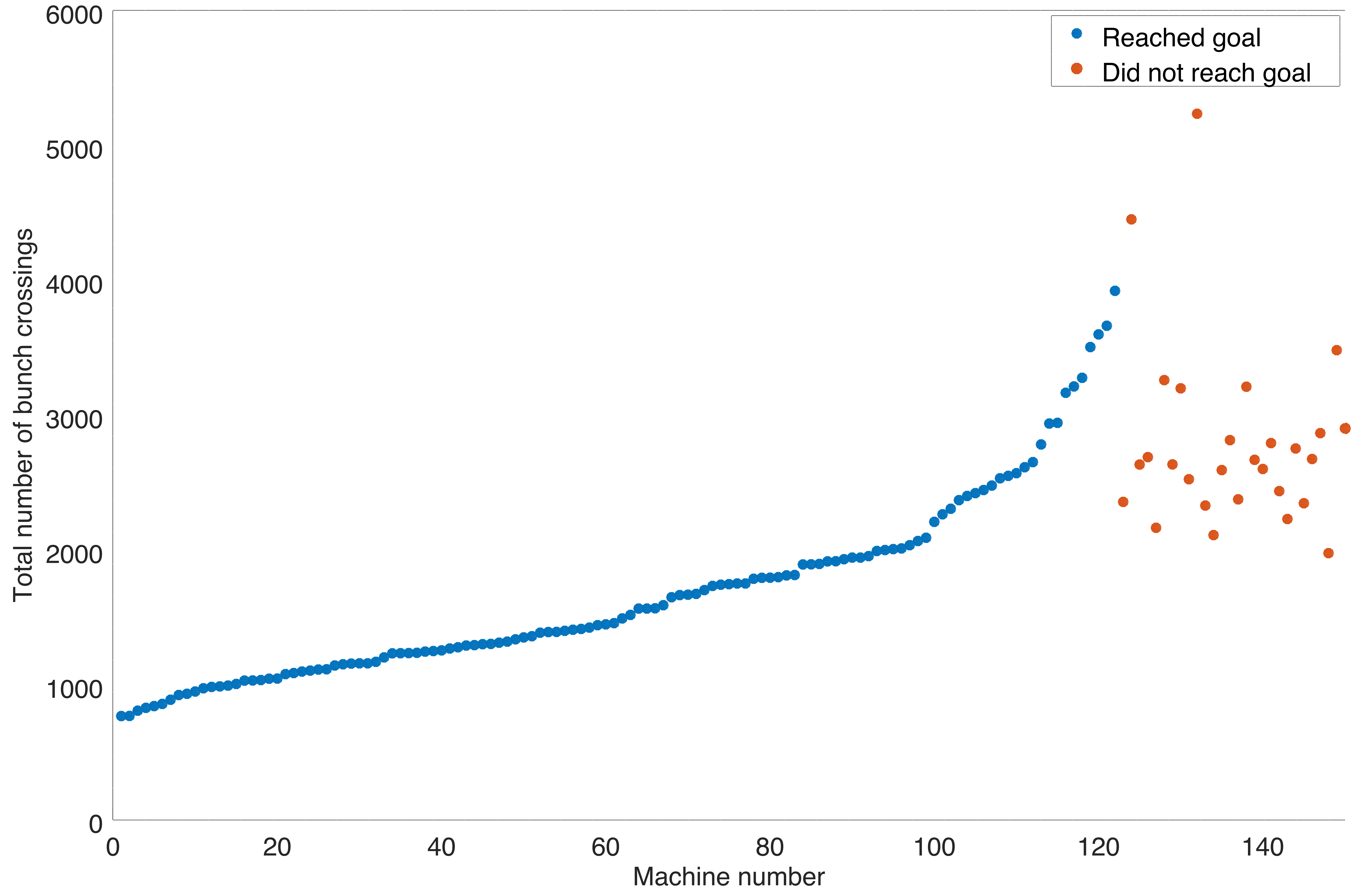
Luminosity histogram

Current state:

- 121 out 150 machines reached tuning goal
- i.e. 81% and not quite 90%
- The worst machine tuned to 81% of target
- How to fine-tune? Signal from pairs is noisy



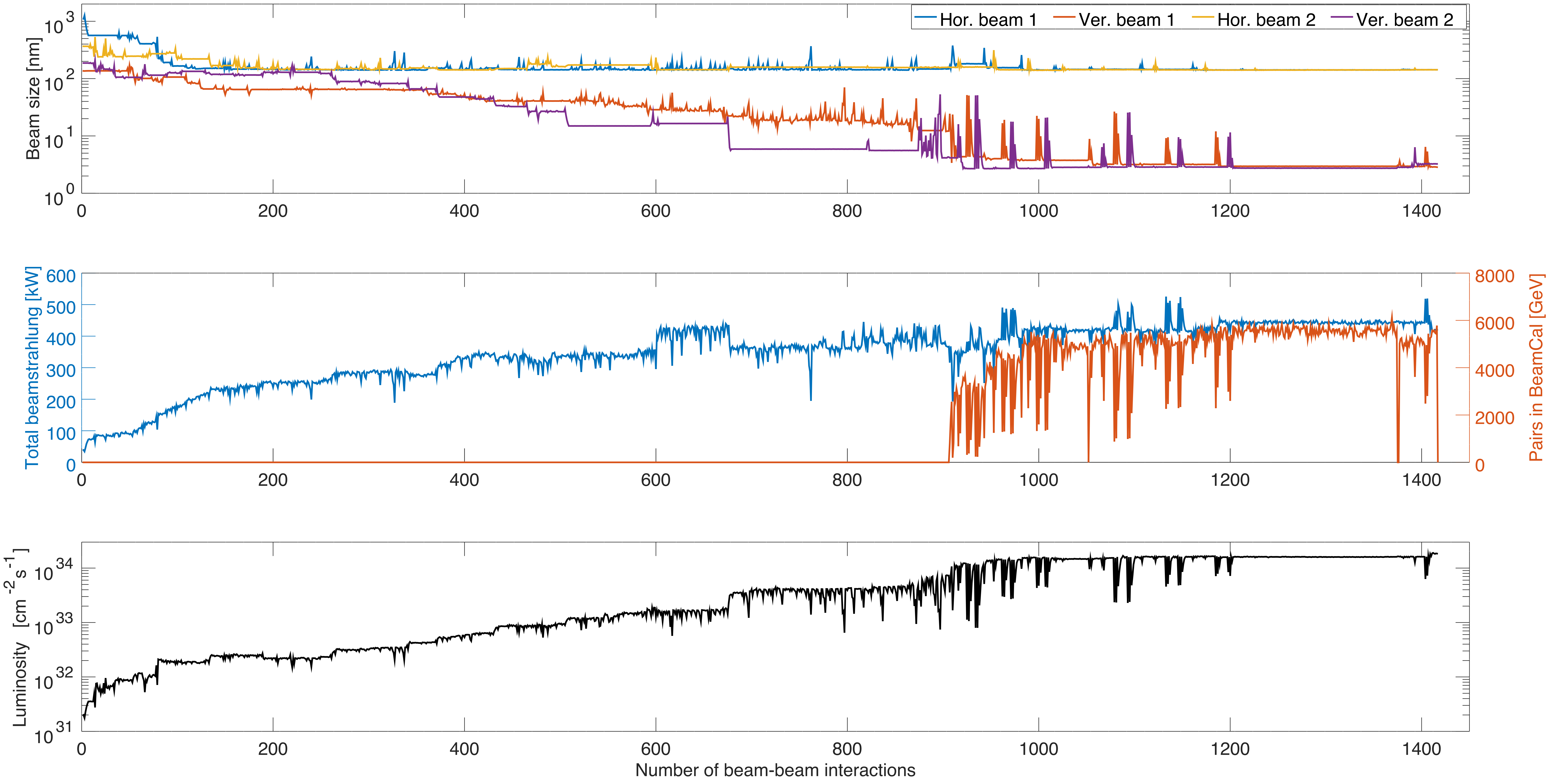
Histogram: number of iterations



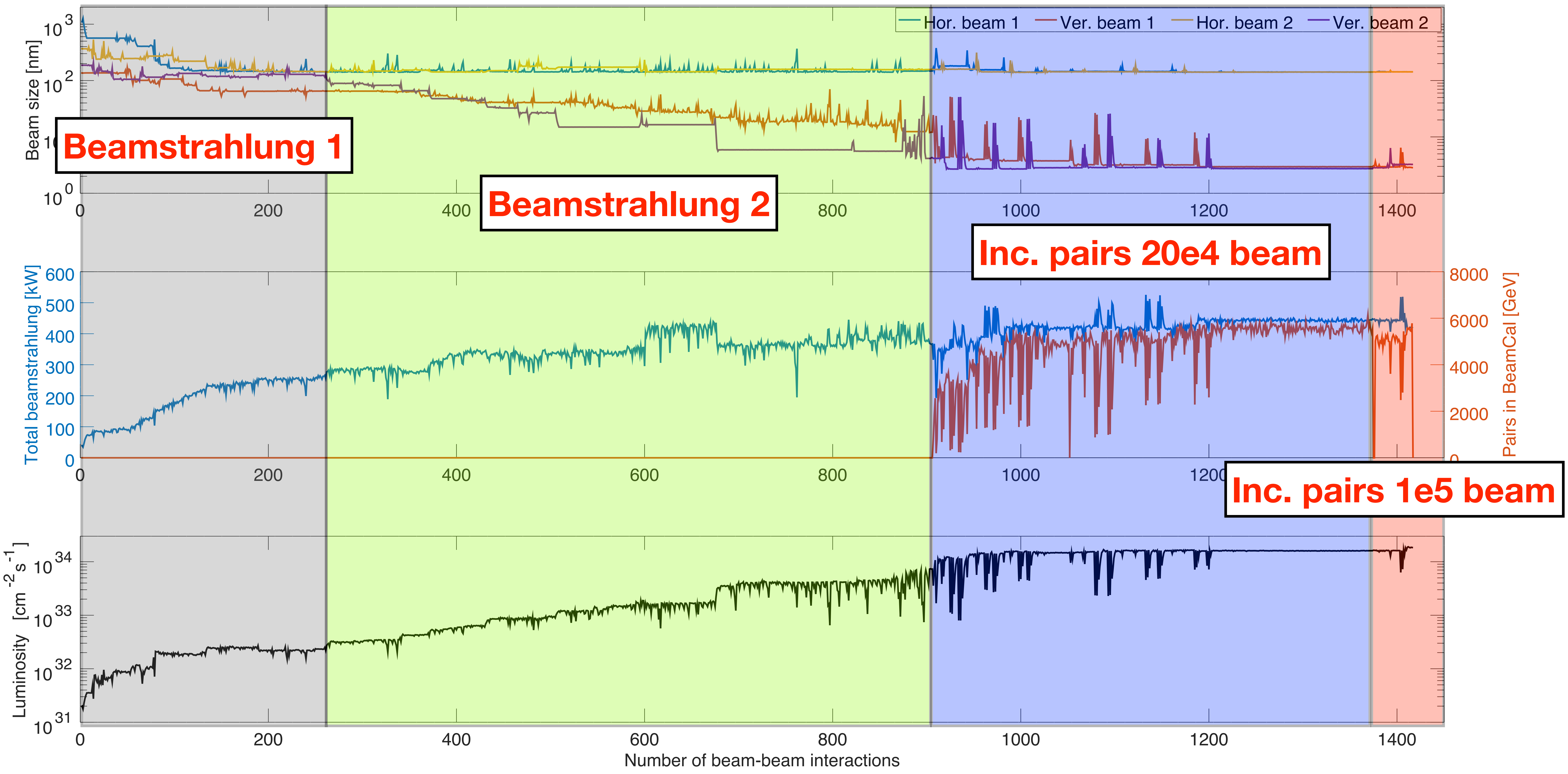
N_{min}	729
N_{max}	3922
N_{mean}	1675
N_{median}	1476

- Mean number of iterations is less than 2x900 (single-beam)
- Not all tuned yet
- Knob scans stopped as target was reached

Tuning evolution of median machine

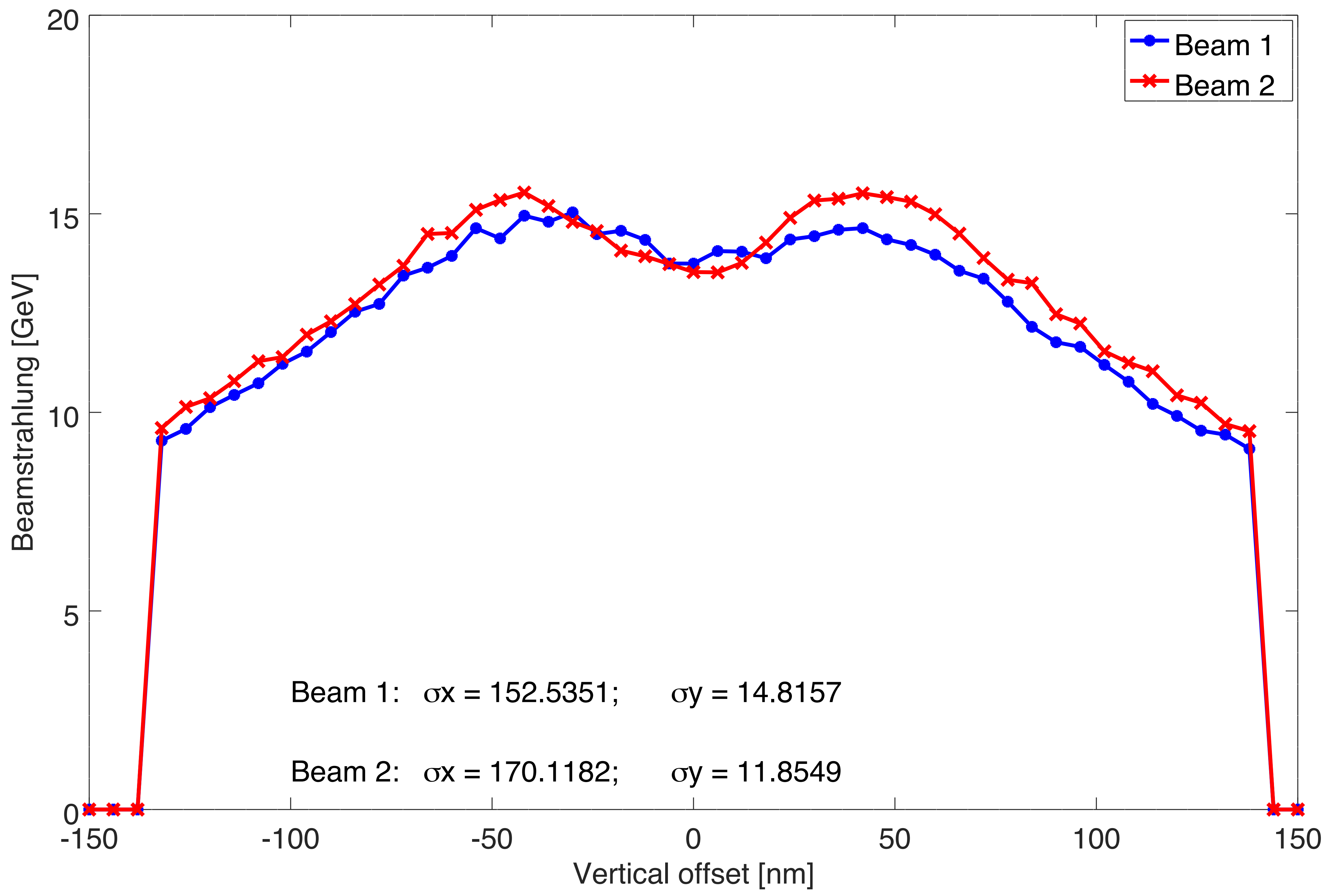


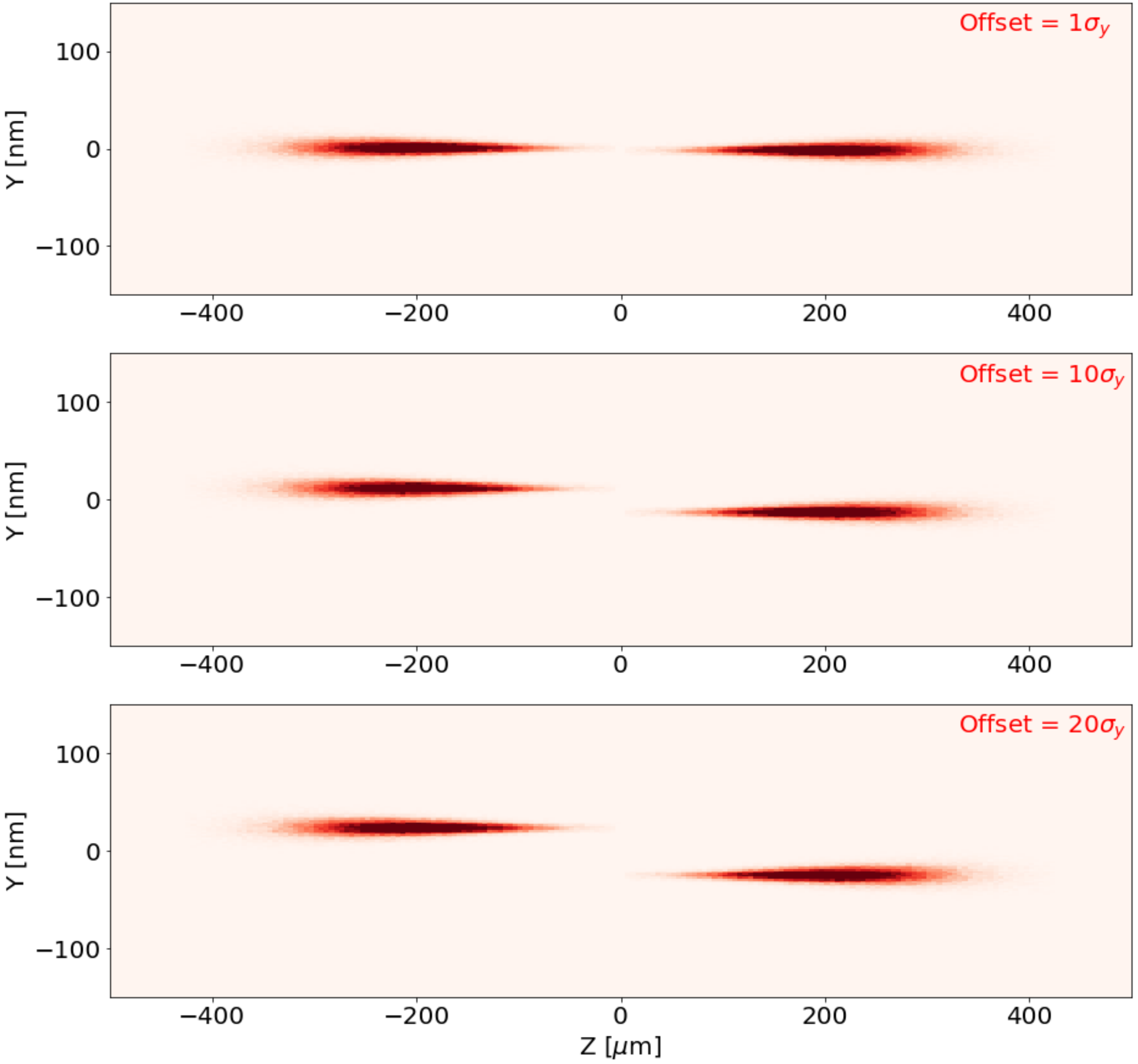
Tuning evolution of median machine

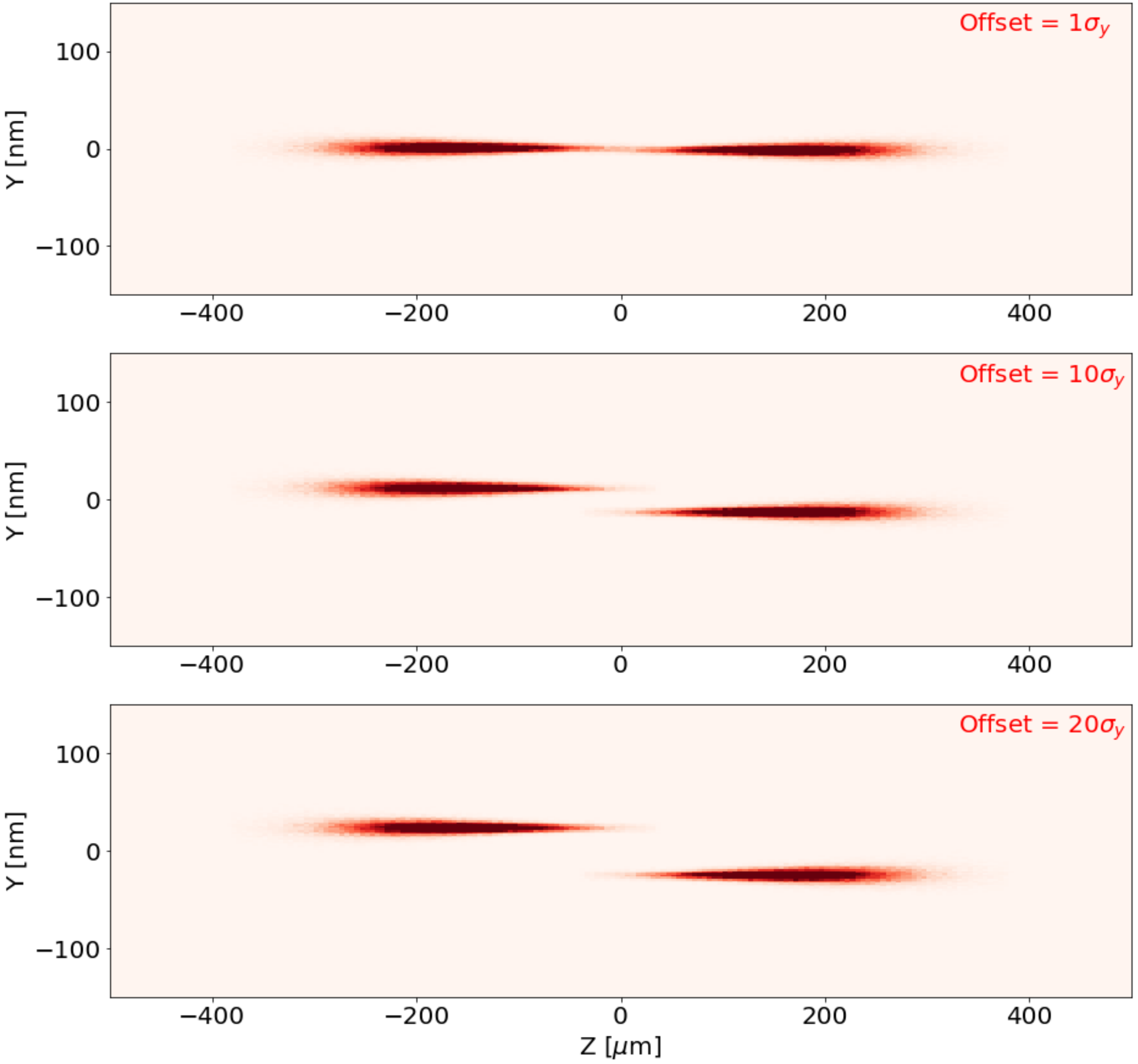


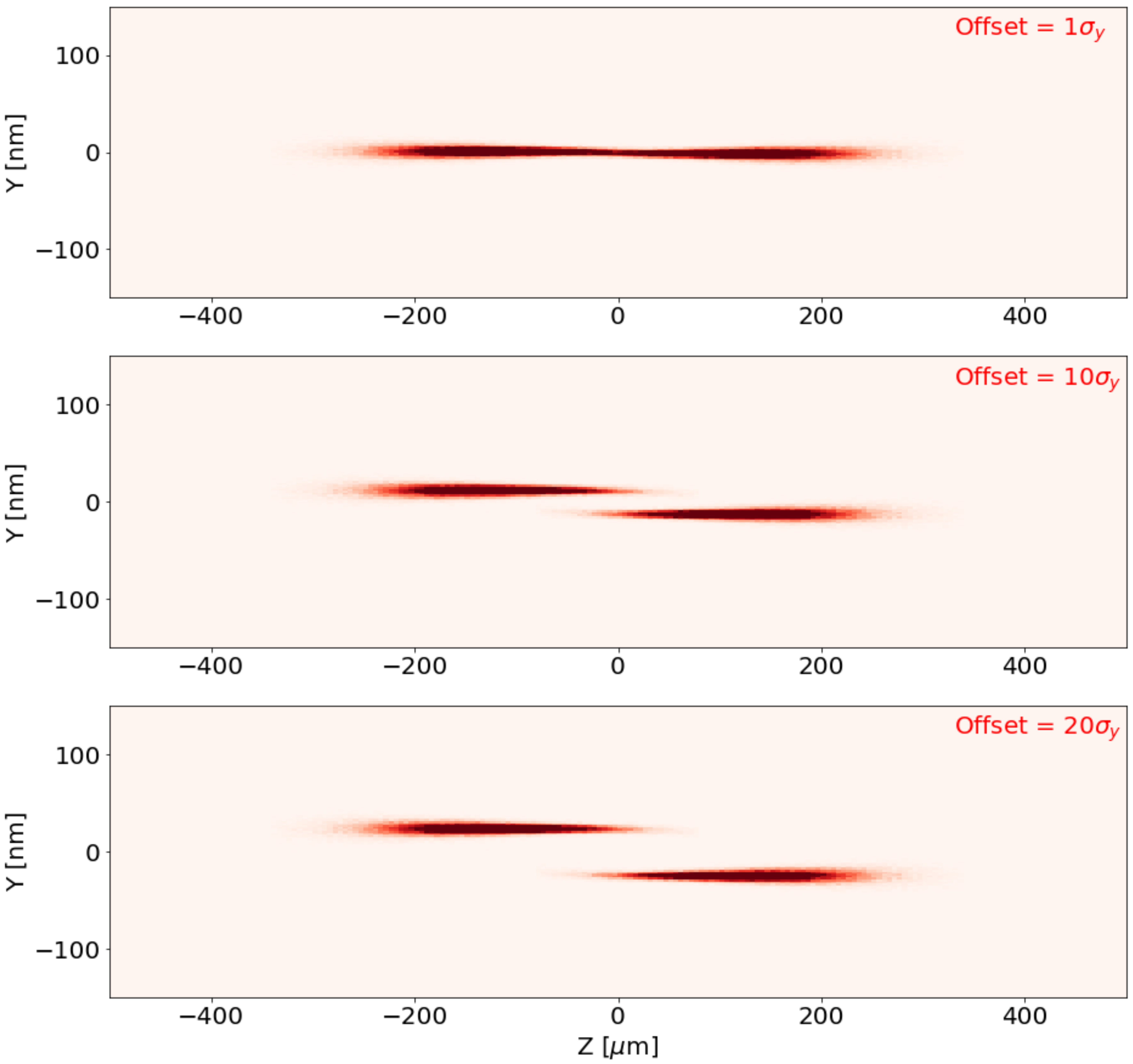
Tuning for vertical beam size?

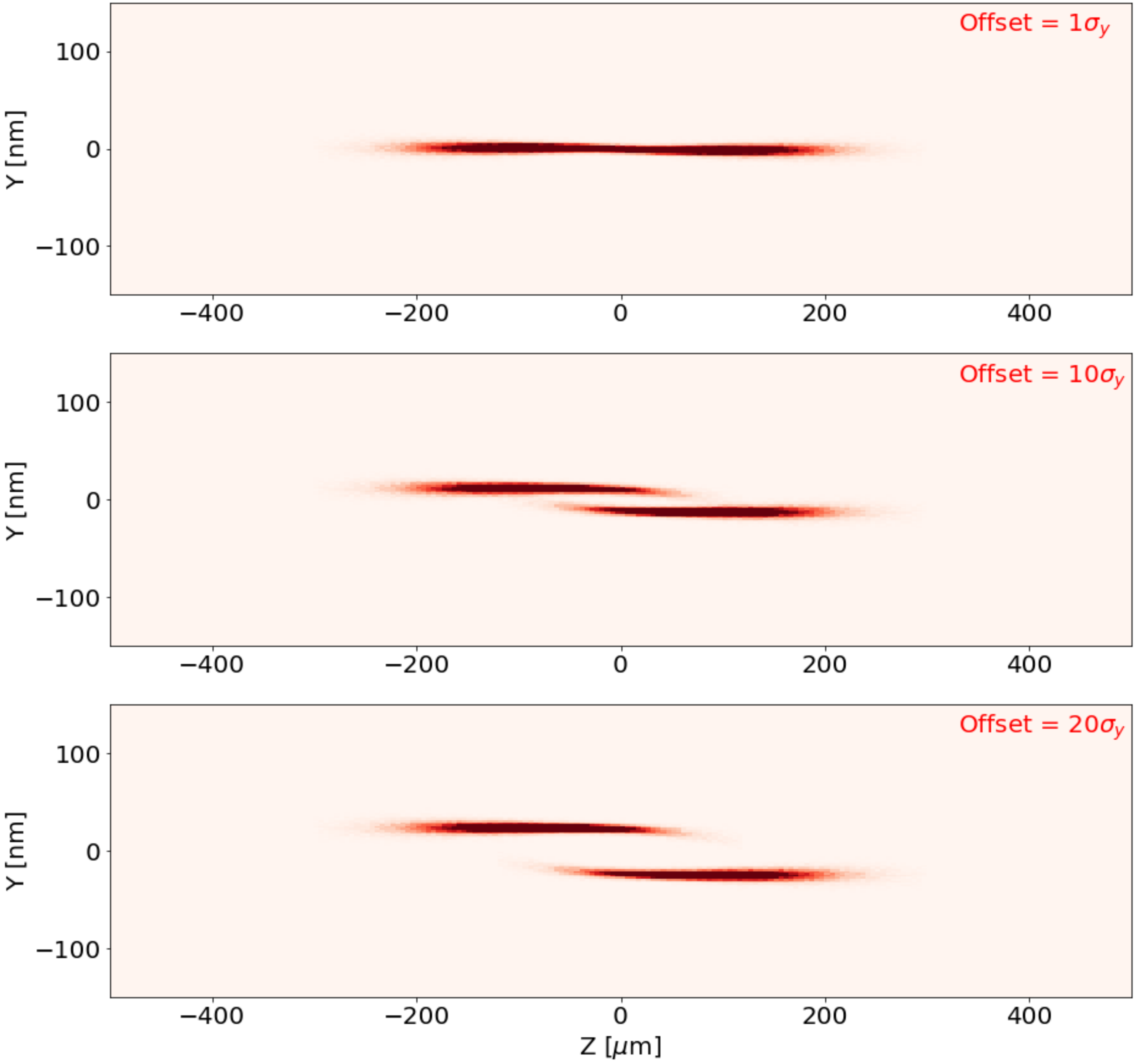
- If photon beam size cannot be measured, we need another signal:
- Beam-beam scans:
  - Move one beam vertically
  - Resulting beamstrahlung power depends on beam sizes
  - Used at SLC and compared with analytical expressions
  - For CLIC not possible. ML?
- Use beam-beam scans for vertical beam size
- Simulation issue: time-consuming



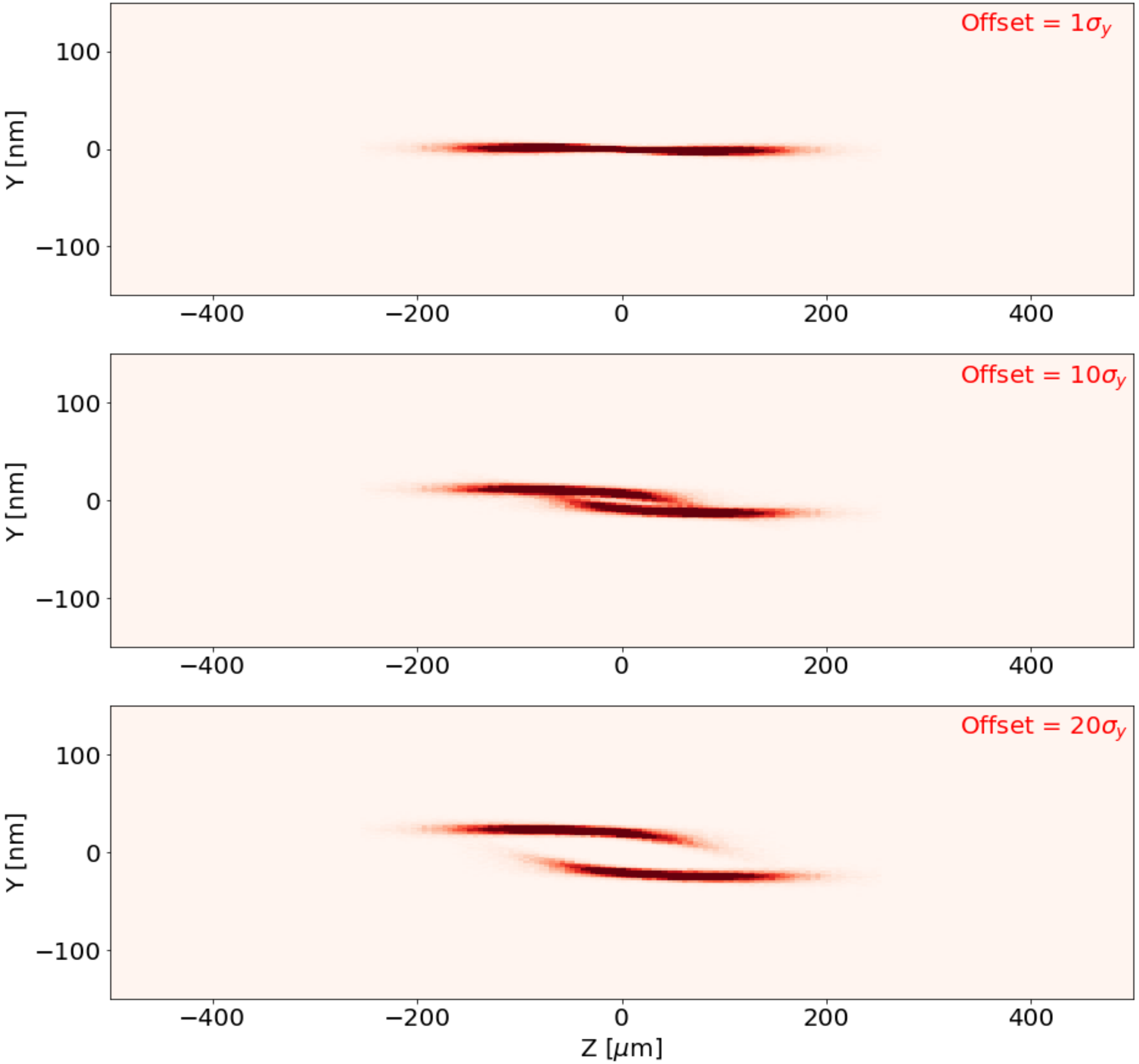

Offset = $1\sigma_y$
Offset = $10\sigma_y$
Offset = $20\sigma_y$
100
0
−100
Y [nm]
−400
−200
0
200
400
Z [μm]

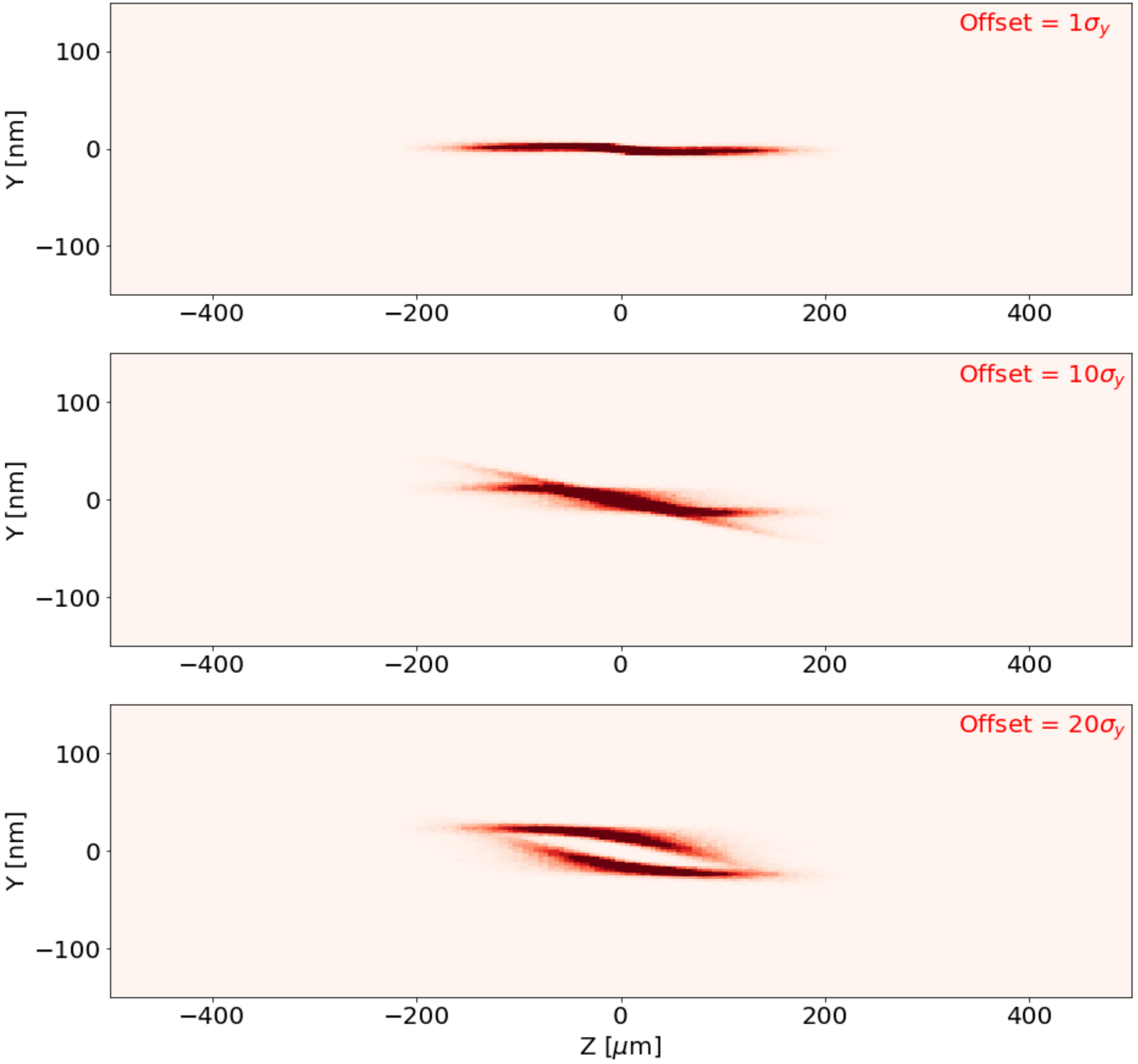

Offset = $1\sigma_y$
Offset = $10\sigma_y$
Offset = $20\sigma_y$
Y [nm]
100
0
−100
−400
−200
0
200
400
Z [μm]

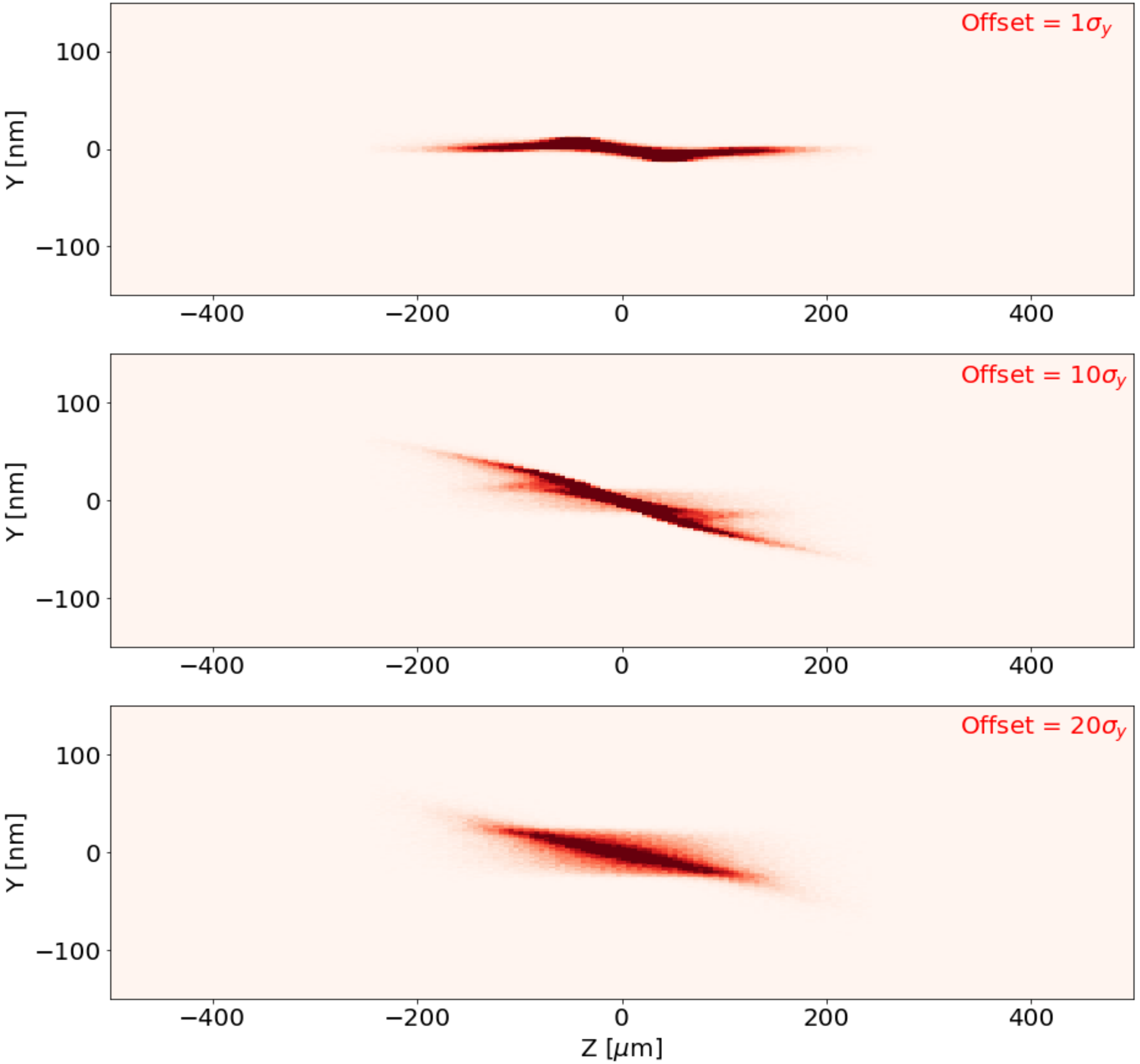

Offset = $1\sigma_y$
Offset = $10\sigma_y$
Offset = $20\sigma_y$
Y [nm]
100
0
−100
−400
−200
0
200
400
Z [μm]

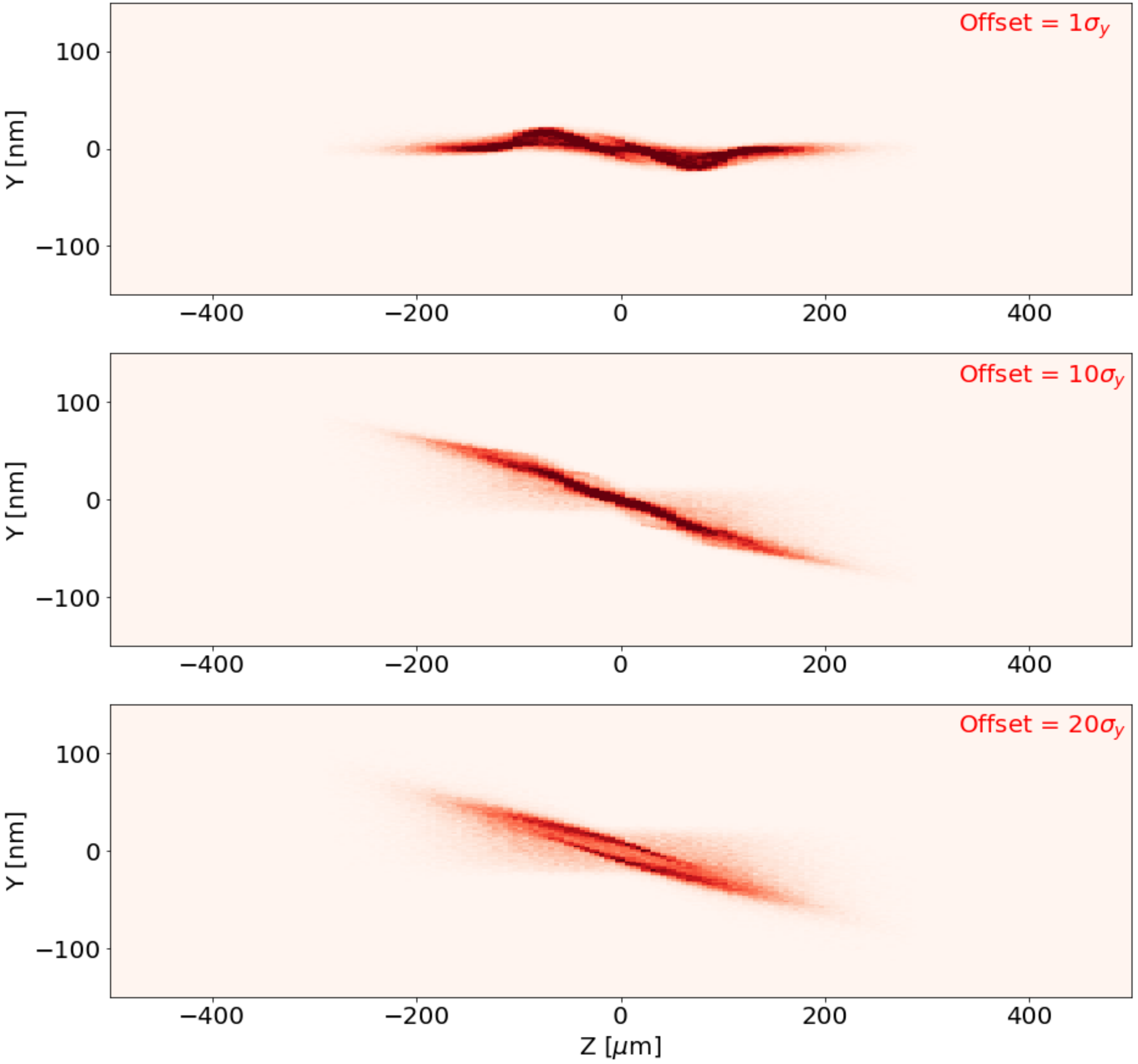


Offset = $1\sigma_y$
Offset = $10\sigma_y$
Offset = $20\sigma_y$
Y [nm]
100
0
−100
−400
−200
0
200
400
Z [μm]

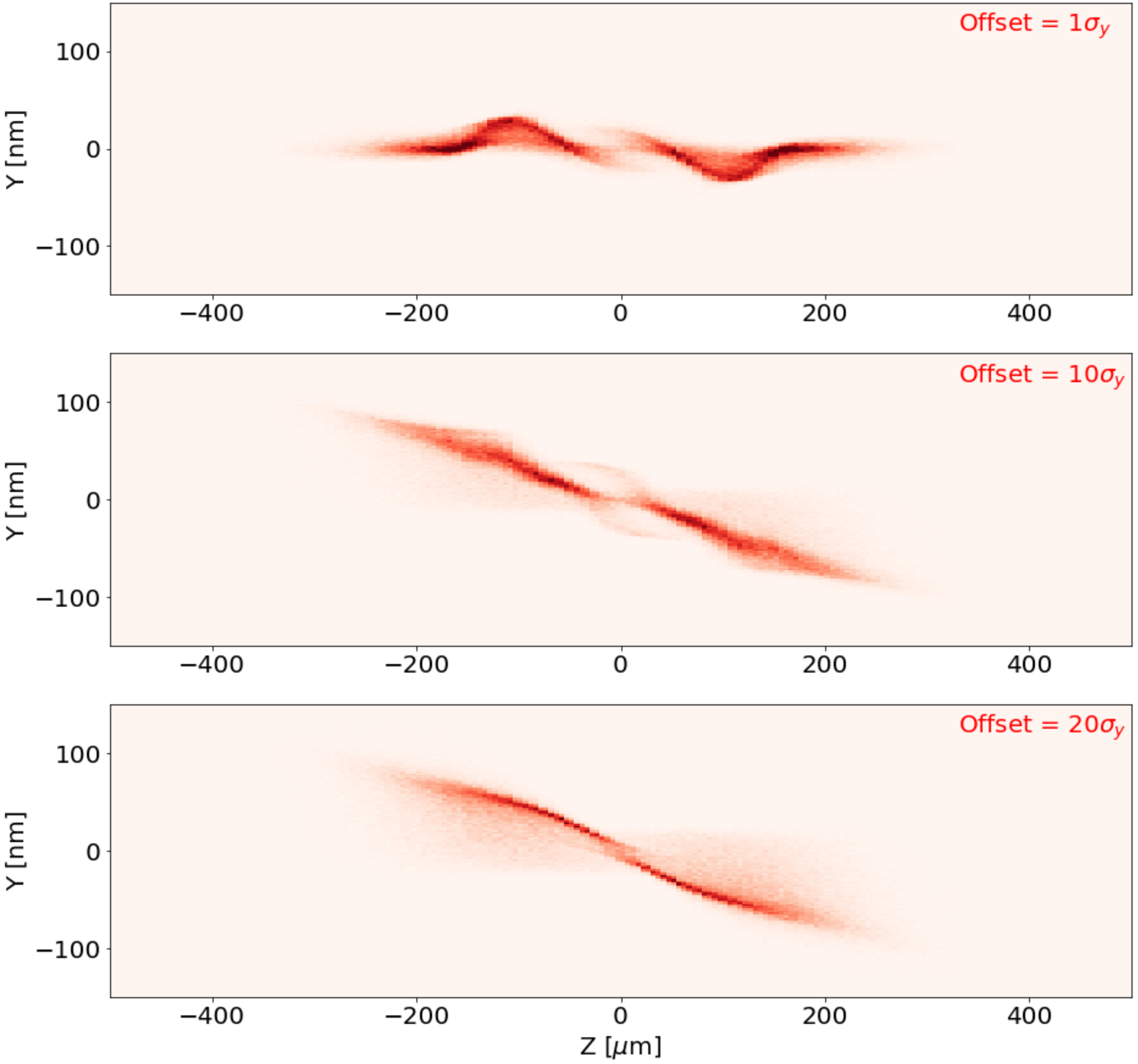

Offset = $1\sigma_y$
Offset = $10\sigma_y$
Offset = $20\sigma_y$
Y [nm]
100
0
−100
−400
−200
0
200
400
Z [μm]


Offset = $1\sigma_y$
Offset = $10\sigma_y$
Offset = $20\sigma_y$
Y [nm]
Z [μm]
100
0
−100
−400
−200
200
400

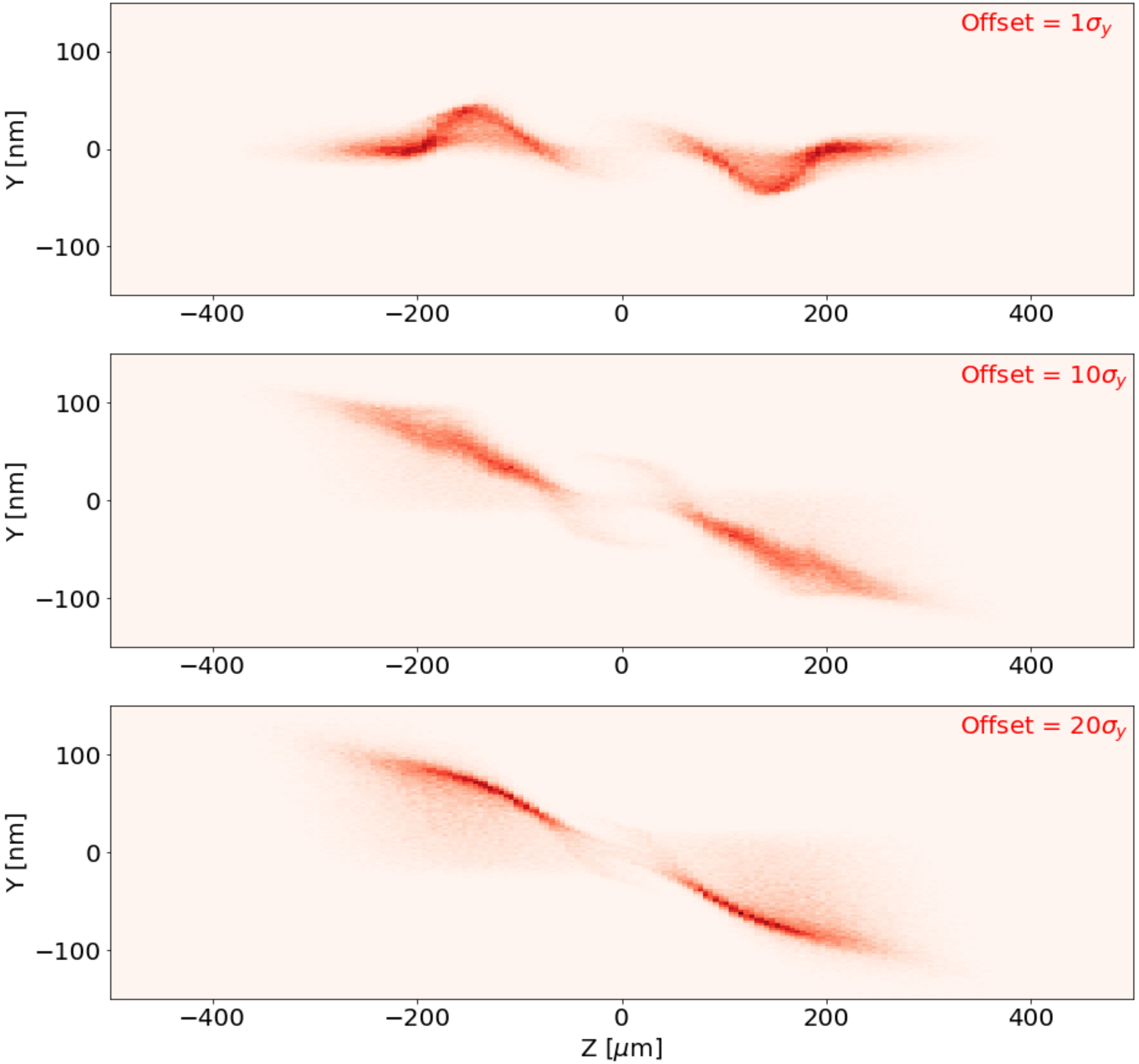

Offset = $1\sigma_y$
Offset = $10\sigma_y$
Offset = $20\sigma_y$
Y [nm]
100
0
−100
−400
−200
0
200
400
Z [μm]

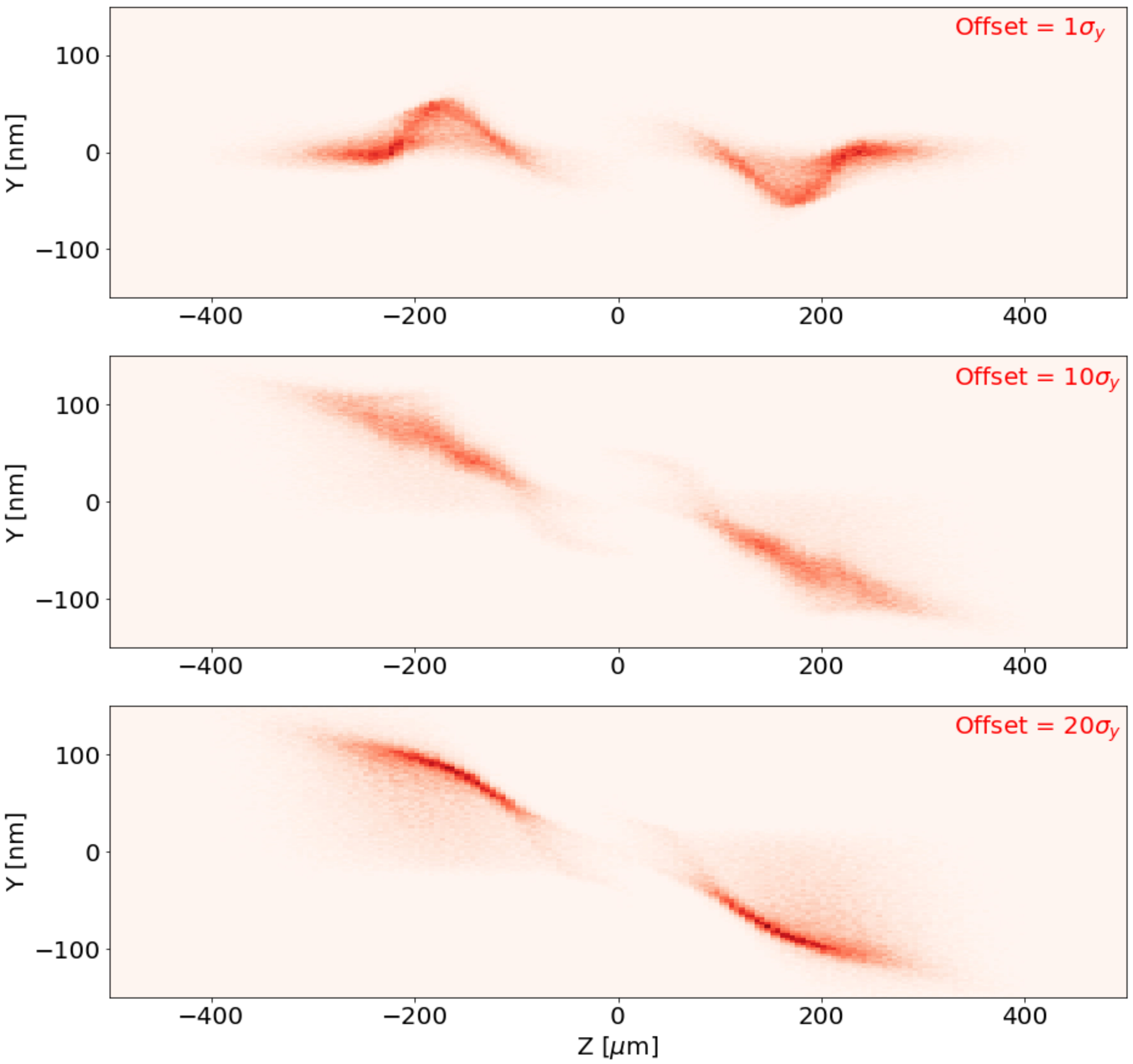


Offset = $1\sigma_y$
Offset = $10\sigma_y$
Offset = $20\sigma_y$
Y [nm]
100
0
−100
−400
−200
0
200
400
Z [μm]



Offset = 1σ_y
Offset = 10σ_y
Offset = 20σ_y
Y [nm]
100
0
−100
−400
−200
0
200
400
Z [μm]


Offset = $1\sigma_y$
Offset = $10\sigma_y$
Offset = $20\sigma_y$
Y [nm]
100
0
−100
−400
−200
0
200
400
Z [μm]



Offset = $1\sigma_y$
Offset = $10\sigma_y$
Offset = $20\sigma_y$
Y [nm]
100
0
−100
−400
−200
0
200
400
Z [μm]

Parallel execution?

- PLACET is a parallelized code but GUINEA-PIG is not
- Tuning with pairs or beam-beam scans: execution time dominated by GUINEA-PIG
- On lxplus or HTCondor normally 10 cores available, GUINEA-PIG utilizes only one
- Can we optimize workflow?
- E.g. beam-beam scan:

Parallel execution?

- PLACET is a parallelized code but GUINEA-PIG is not
- Tuning with pairs or beam-beam scans: execution time dominated by GUINEA-PIG
- On lxplus or HTCondor normally 10 cores available, GUINEA-PIG utilizes only one
- Can we optimize workflow?
- E.g. beam-beam scan:

Track in placet → Add different offsets → Execute sequentially

Track in placet → Add different offsets → Execute in parallel

Parallel execution?

- PLACET is a parallelized code but GUINEA-PIG is not
- Tuning with pairs or beam-beam scans: execution time dominated by GUINEA-PIG
- On lxplus or HTCondor normally 10 cores available, GUINEA-PIG utilizes only one
- Can we optimize workflow?
- E.g. beam-beam scan:

Track in placet → Add different offsets → Execute sequentially

Track in placet → Add different offsets → Execute in parallel

Parallel execution can easily be done in bash:

Sequential for:

```
for i in {…}; do
    command ;
done
```

Parallel for:

```
for i in {…}; do
    command &
done
```

Implementation

```
proc test_electron_BB { step_size } {
     CHANGE KNOB SETTING
     TRACK AND SAVE BEAMS
     exec bash -c "for i in `seq -450 50 450`  ; do ./run_guinea_beam_scan_y.sh \$i & done"
     LOAD AND ANALYZE RESULTS
}
```

Implementation

```
proc test_electron_BB { step_size } {
     CHANGE KNOB SETTING
     TRACK AND SAVE BEAMS
     exec bash -c "for i in `seq -450 50 450`  ; do ./run_guinea_beam_scan_y.sh \$i & done"
     LOAD AND ANALYZE RESULTS
}
```

run_guinea_beam_scan_y.sh

```
#!/bin/sh
mkdir -p temp_$1
cd temp_$1
cp ../run_guinea_beam_scan_y.tcl .
placet run_guinea_beam_scan_y.tcl offset_ver $1
cd ../
rm -rf temp_$1
```

Implementation

```
proc test_electron_BB { step_size } {
     CHANGE KNOB SETTING
     TRACK AND SAVE BEAMS
     exec bash -c "for i in `seq -450 50 450`  ; do ./run_guinea_beam_scan_y.sh \$i & done"
     LOAD AND ANALYZE RESULTS
}
```

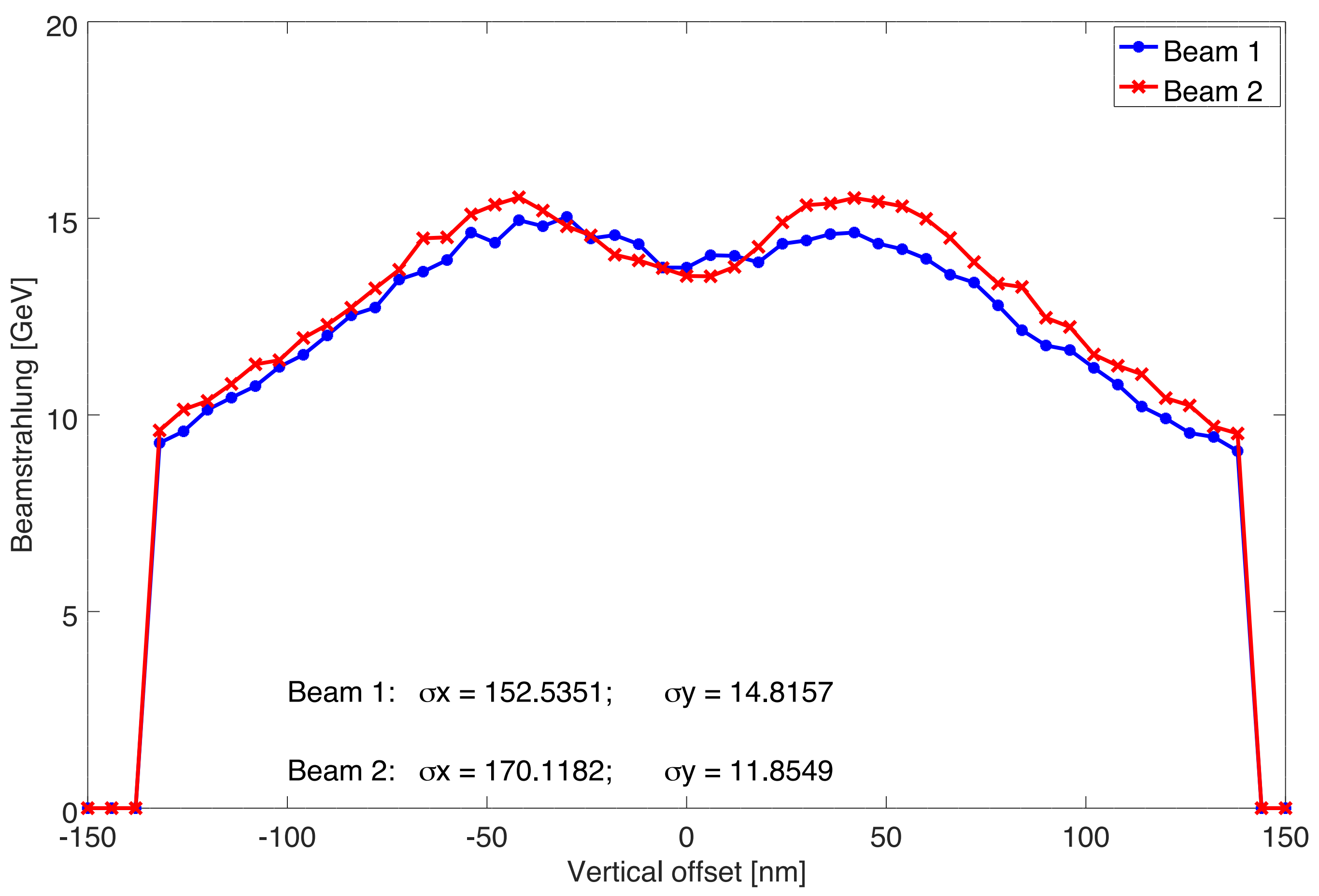
run_guinea_beam_scan_y.sh

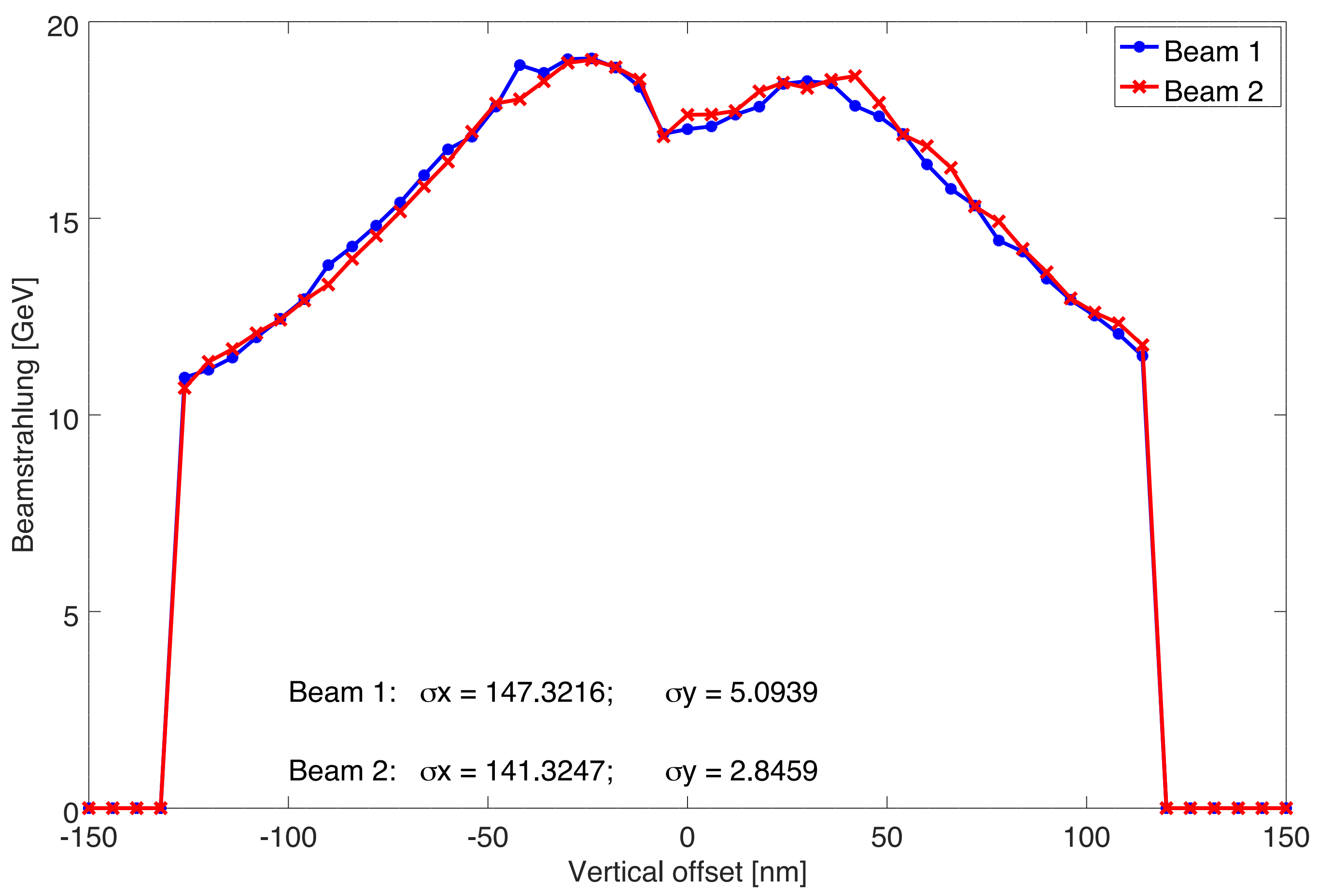
```
#!/bin/sh
mkdir -p temp_$1
cd temp_$1
cp ../run_guinea_beam_scan_y.tcl .
placet run_guinea_beam_scan_y.tcl offset_ver $1
cd ../
rm -rf temp_$1
```

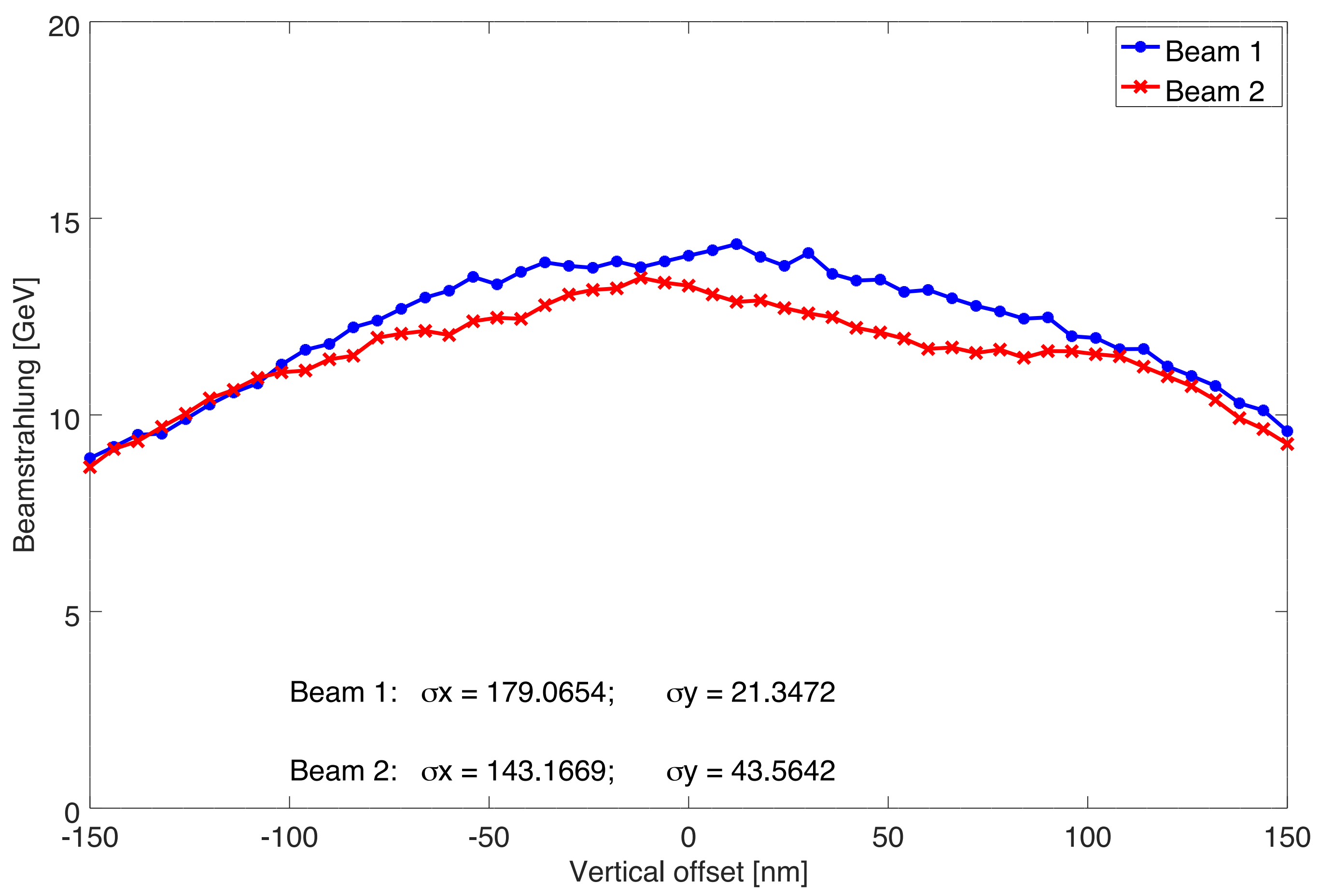
run_guinea_beam_scan_y.tcl

```
SETUP GUINEA-PIG PARAMETERS
…
source $script_dir/scripts/calc_lumi_jim.tcl

Octave {
   B1 = load("../beams_temp/electron.dat");
   B2 = load("../beams_temp/positron.dat");

   # add vertical offset
   B1(:,3) = B1(:,3)+${offset_ver}*1e-3*ones(size(B1(:,3)));
   [L, Lpeak, beam_size, beam_strahl, beam_defl] = get_lumi(B1, B2, 1)
   save ../GP_output_temp/GP_output_${offset_ver}.dat L Lpeak beam_size beam_strahl beam_defl
}
```

Tuning σ_y using beam-beam scans







- First order signal:

$$Y = \frac{\mathbf{max}(f)}{\mathbf{FWHM}(f)}$$

- Works well when beams are above a certain size
- Attempted in tuning, replacing *Beamstrahlung 2*
- Also tried: pattern recognition using neural networks (just started…)

Two-beam tuning current status

- Using beamstrahlung for initial tuning works well
- Tuning vertical beam size using beam-beam scan seems promising
    - Successful in replacing old method in a few cases
    - Needs more work
    - Improve on robustness of parallel execution of GP

- Fine-tuning using the pairs is difficult
    - Noisy signal and tuning does not always improve
    - Perhaps using a combination of signals?

- Close to reaching the goal
- Quite fast tuning (700-4000) beam-beam interactions

Conclusions

- Double beam tuning with realistic signals
  - Random walks with beamstrahlung and knob scans with incoherent pairs
  - Tracking pairs is slow and tuning simulations become time-consuming
- Current state: 81% of machines reached tuning goal of 110% of L_0
- One-beam vs. two-beam tuning
  - One-beam can give stronger signal due to symmetric change
  - Two-beam is less likely to get stuck in local optimum since the other beam changes
- Challenges
  - Information of vertical photon beam size might not be measurable
  - Pairs signal quite noisy and difficult to use for fine-tuning
- Future work
  - Test beam-beam scans for vertical beam size tuning
  - Parallel execution of GP to speed up run time
  - Test combination of signals for fine-tuning
  - Use beams from integrated simulation (including upstream tuning)
  - Horizontal beam size limit?
  - Machine learning?
  - …