Beam-Beam Simulations at SLAC

Andreas Kabel

SLAC/ACD

4/9/2009

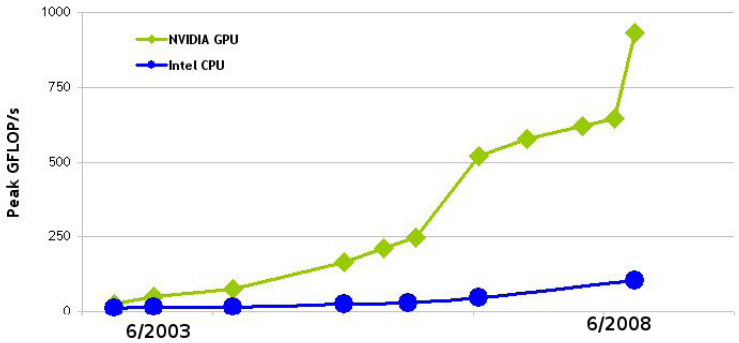
Computational Challenges

- Computationally:
  - LHC: 4 IPs, 120 parasitics
  - Non-linear lattice, 1000s of elements
- "Predictive power": can one go beyond DA?
- Beam lifetimes, emittance growth, ...
- Code + computing capabilities approach realistic time scales
- Appreciable differences in approximations used

The PLIBB Code

- Used for Tevatron, RHIC, LHC
- Full tracking code, highly parallel
  - MAD-X lattices
  - handles all elements in RHIC, LHC lattices
  - Embedded scripting language
- Fully 6D throughout, uses differential algebra
- Creates fast tracking kernels in different approximations (Thick lenses, thin lenses: drift-kick, 4th order, 6th order)
- Symplectic lumping: absorb quadrupoles + bends between interesting elements into single 3rd order integrable Hamiltonian
- Weak-strong beam-beam, current wire, electron lens (multi-bunch, soft gaussian, strong-strong)

Particle Tracking on Graphics Cards (with D. Yershov)



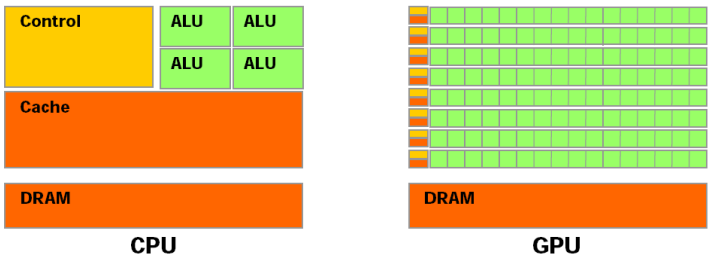


Figure 1-2. The GPU Devotes More Transistors to Data Processing

GPUs:

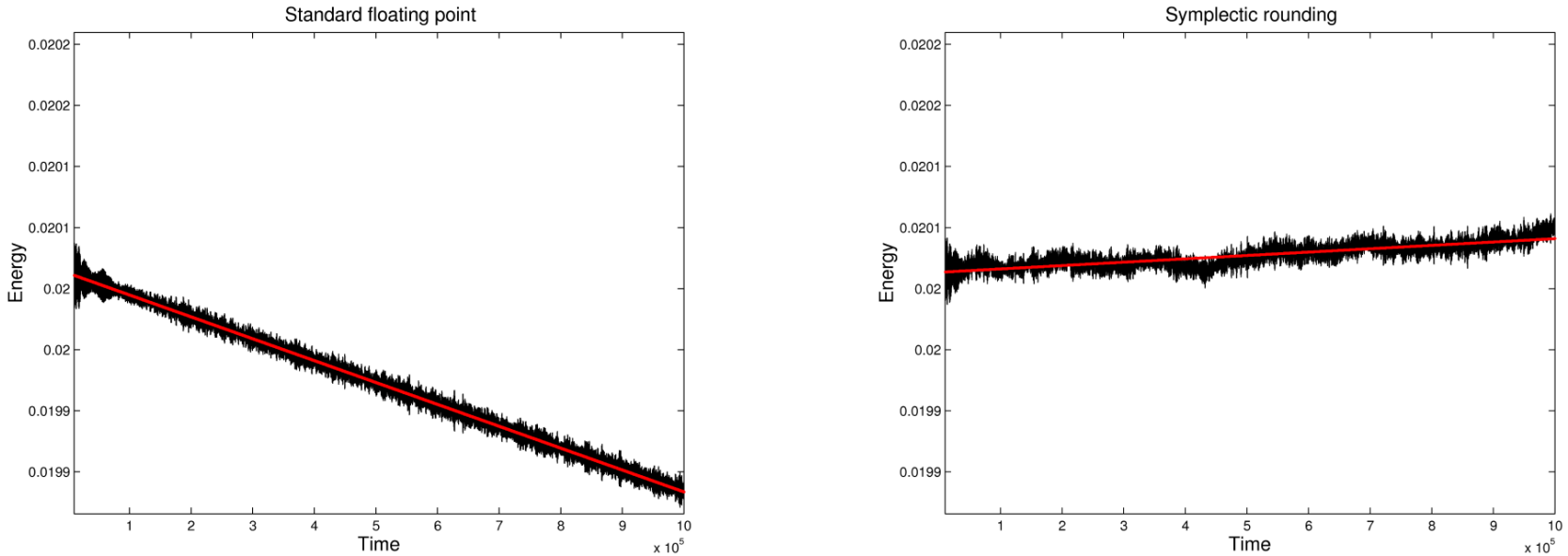
- Many ALUs, many lightweight threads
- Maximum benefir for data-parallel, branchless code (ray tracing, texture mapping...)
- Beam dynamics: Ray tracing in 6 dimensions

Symplectification

- Symplectic integrators: correct integration of a 'nearby' Hamiltonian.
- Roundoff errors will introduce non-Hamiltonian terms, even with symplectic integrators.
- Integers: Lower accuracy, no rounding error
- Resolution required $\sqrt{\frac{\varepsilon_N}{\lambda_{\mathrm{Compton}}}}$
- (Skeel, 1999:) Pinning ν-bit mantissa floating-point number to a $2^\mu \otimes 2^\mu$ lattice in $[-q_{max}, q_{max}] \otimes [-p_{max}, p_{max}]$

$$
\begin{aligned}
{}_q &:= (q + (.75 \cdot 2^{\nu-\mu} q_{max})) \\
p^{n+\frac{1}{2}} &= p^n + [\Delta F^n]_p \\
q^{n+1} &= q^n + \left[\frac{2\Delta}{m} p^{n+\frac{1}{2}}\right]_q
\end{aligned}
\tag{1}
$$

Symplectic Lattice: Toy Example

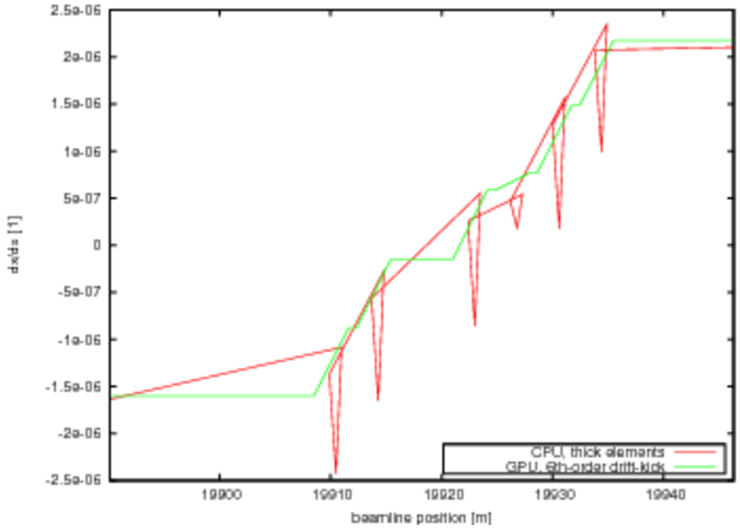


Kinetic energy vs. turn number for plain and lattice single-precision arithmetic (Simplified model: Storage ring in 4×4 smooth approximation, single $1/r$ ('current wire') perturbation)

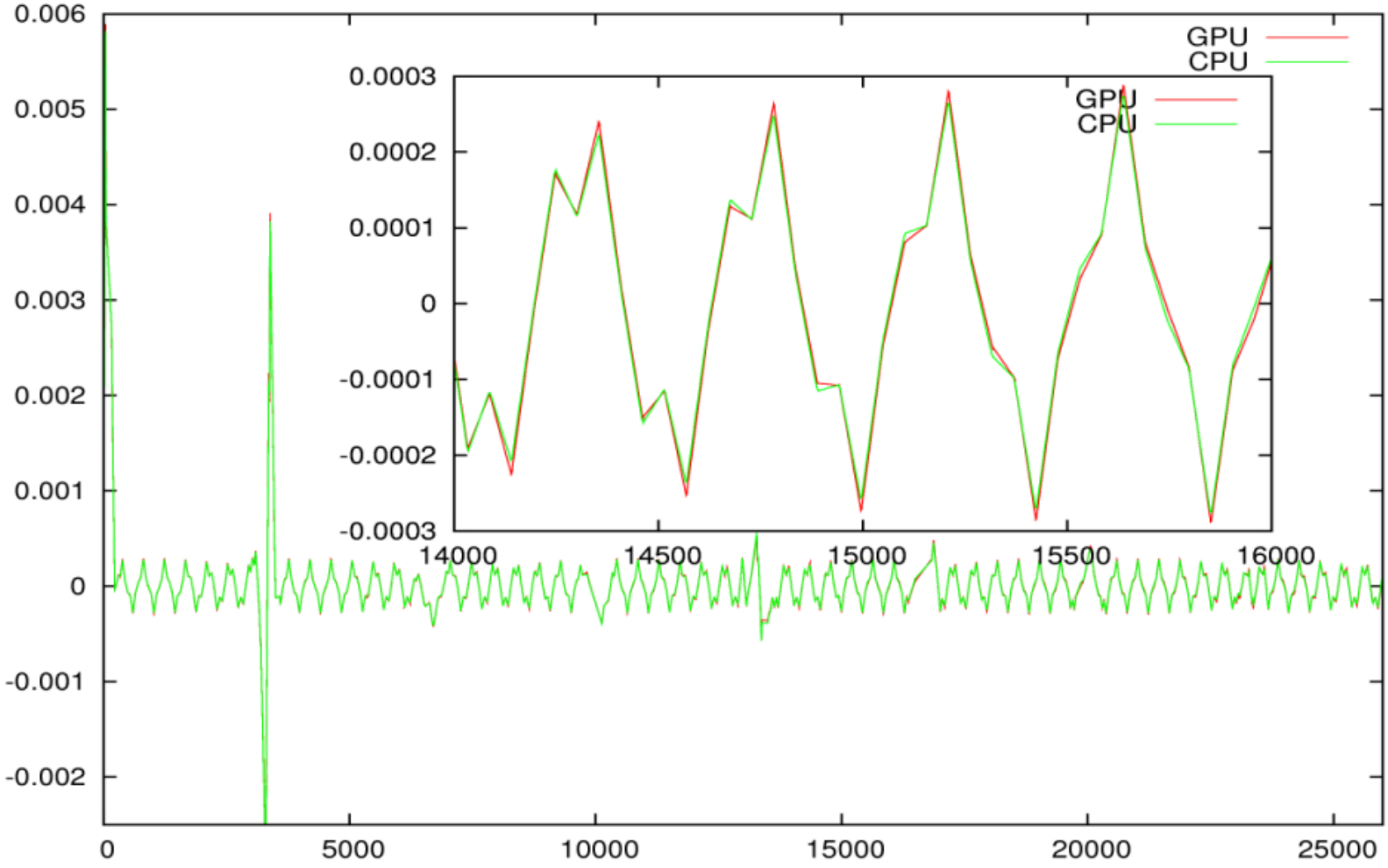
gputrack

PLIBB module gputrack

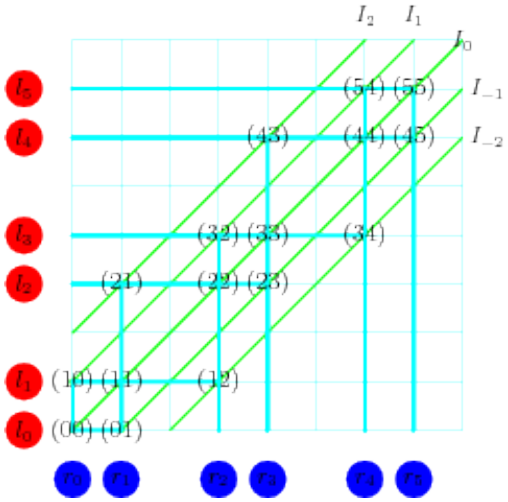
- general, table-driven Drift/Kick kernel running on GPU
- Leapfrog, 6th + 8th order Yoshida implemented
- PLIBB precalculates coefficients
- LHC fits onto GPU texture memory
- Tracks particles in batches of 128
- Full speed gain realized



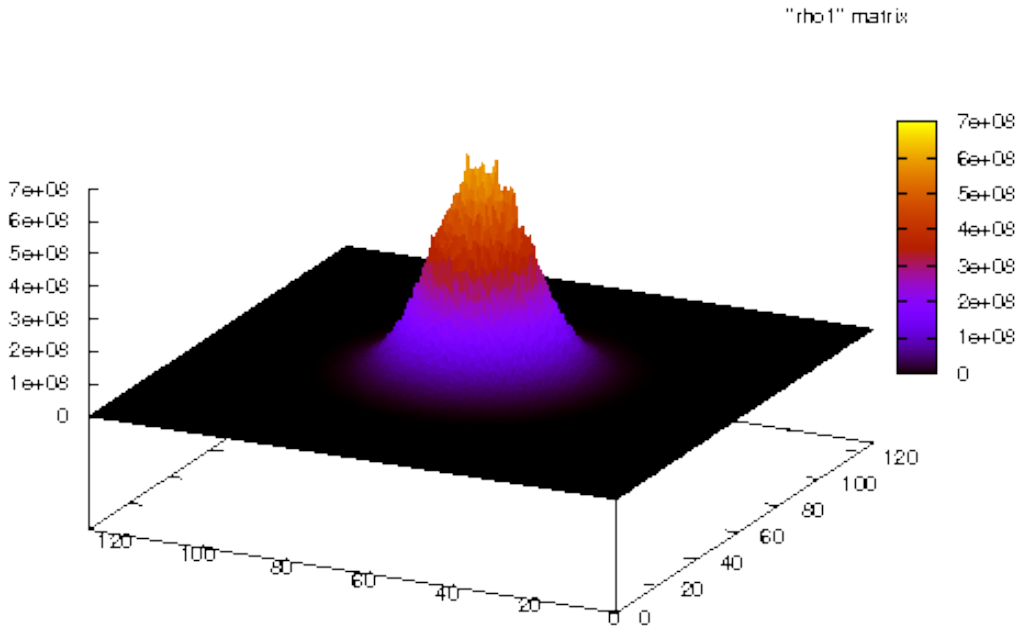
Tracking Through the LHC Lattice

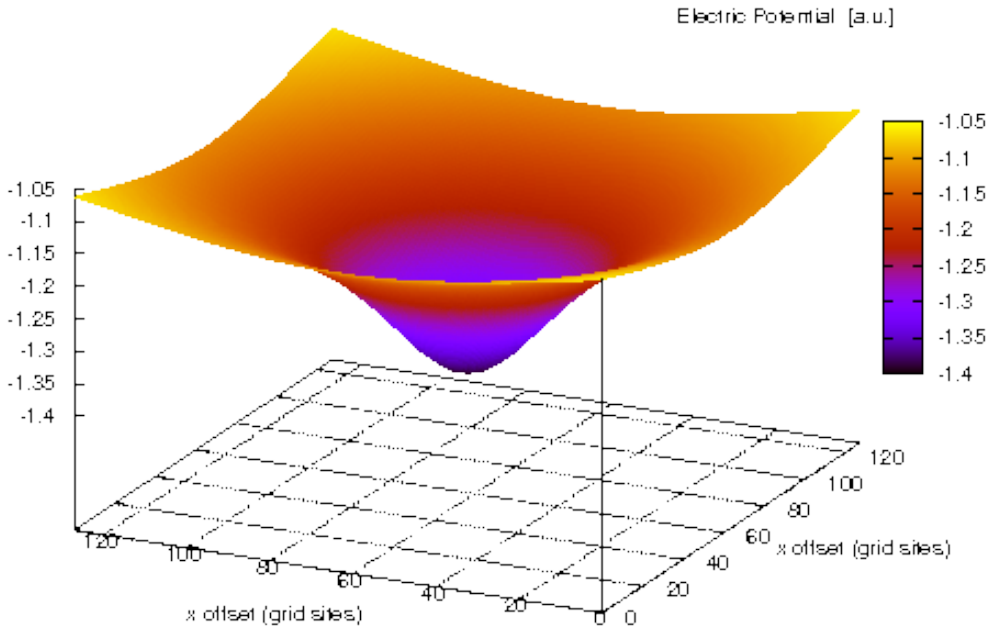


Multi-Bunch



- Run `plibb` one CPU/bunch/beam,
- put 'trap door' elements in IPs
- only pairwise communication needed
- sequence is fixed by communicators
- pipelining, synchronized by update of
  - Full strong-strong: ρ, φ (expensive: 2D convolution solver in PLIBB)
  - $x_{avg}, y_{avg}, \sigma_x, \sigma_y$


"rho1" matrix
7e+08
6e+08
5e+08
4e+08
3e+08
2e+08
1e+08
0
120
100
80
60
40
20
0


Electric Potential [a.u.]
-1.05
-1.1
-1.15
-1.2
-1.25
-1.3
-1.35
-1.4
120
100
80
60
40
20
0
x offset (grid sites)
x offset (grid sites)

Electron Lens Tracking Studies

- Model used: LHC 7TeV, IPs 1,2,5,8
- 8/30 parasitic crossings per IP
- design emittances
- $N = 1.05 \cdot 10^{11}$, $N = 1.3 \cdot 10^{11}$, $N = 1.7 \cdot 10^{11}$,
- Position single electron lens at location, $s = -103m$ (β crossing)
- $\sigma_x \approx \sigma_y \approx 1.02\mathrm{mm}$

Possible E-Lens Configuration

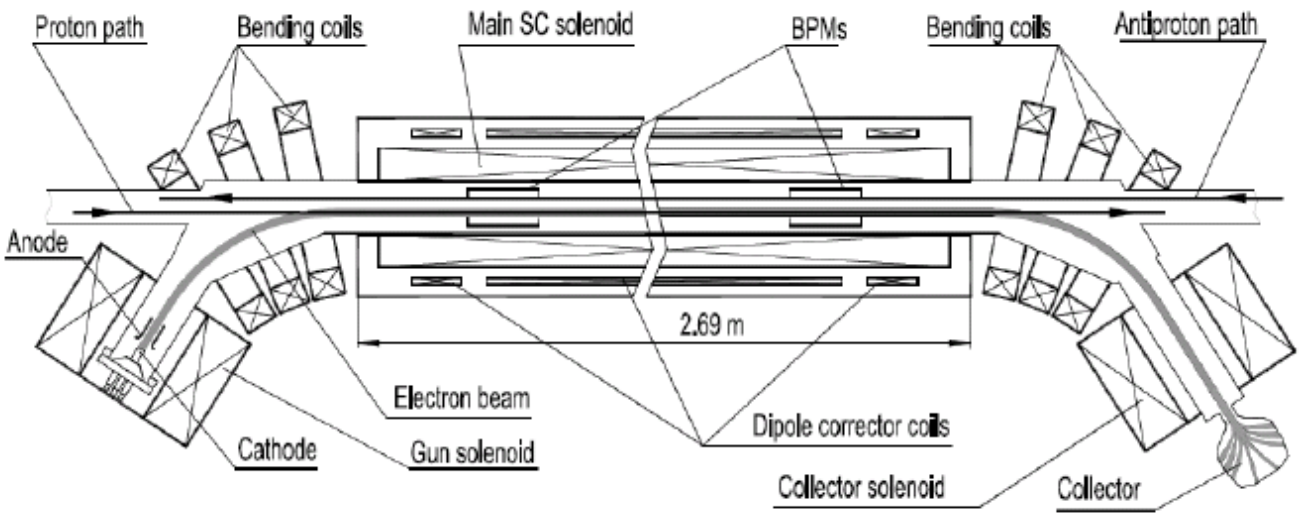


Figure 1: Electron lens configuration for compensation of beam-beam effects.

Electron Lens Element

Modeled as current distribution $I(r)r\mathrm{d}r$ within R (possibly confined by solenoidal external fields)

$$\delta\mathbf{x}'_\perp = k\frac{I_{encl}\mathbf{r}_\perp}{r_\perp^2}, r_\perp > R$$

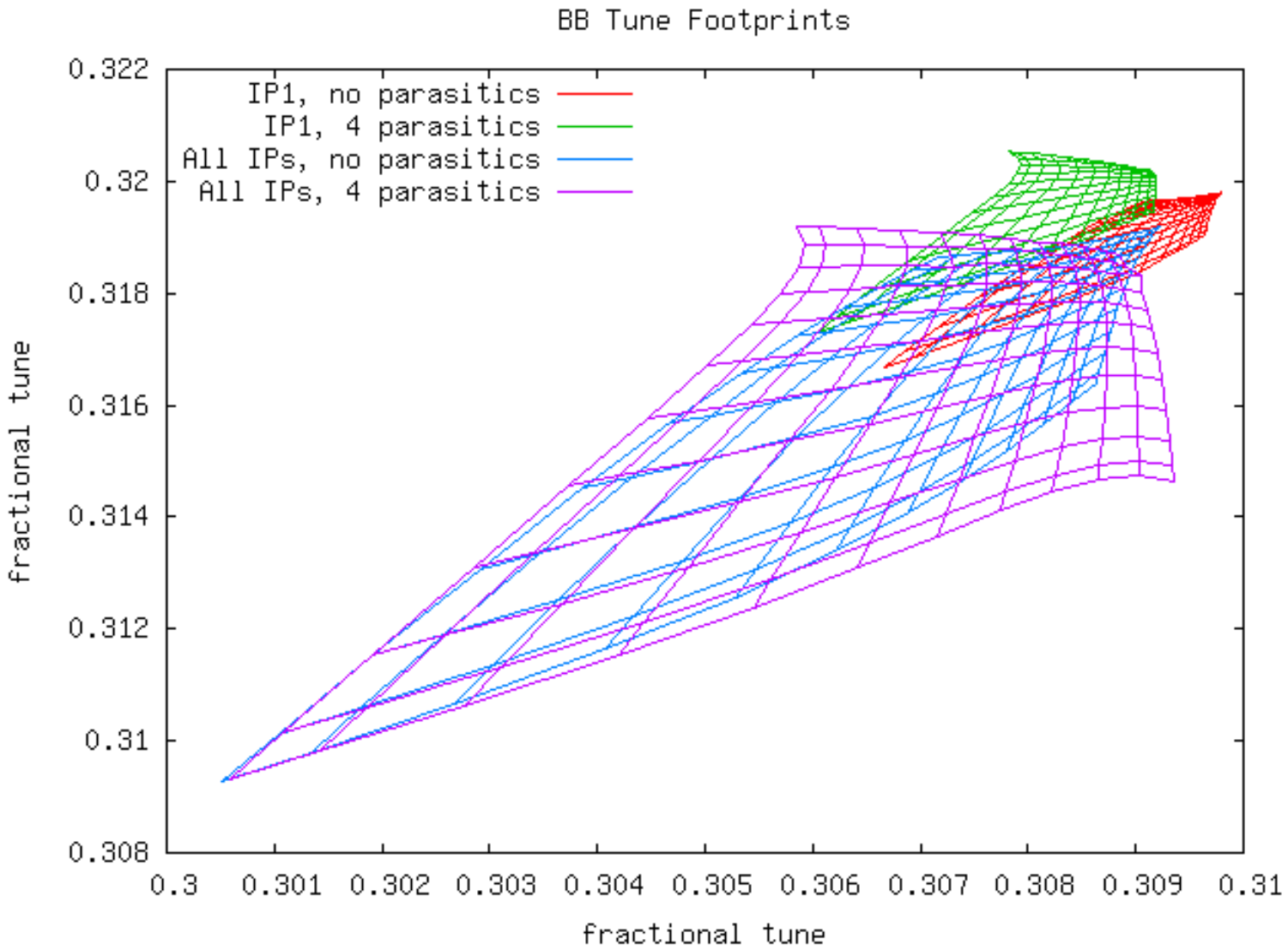
$$\delta\mathbf{x}'_\perp = 2\pi k\mathbf{r}_\perp r_\perp^2 \int_0^{r_\perp} I(r')r'\mathrm{d}r', r_\perp \leq R$$

$$k = \frac{e_p\mu_0}{\gamma_p}\left(1 \pm \frac{1}{\beta_e}\right)$$

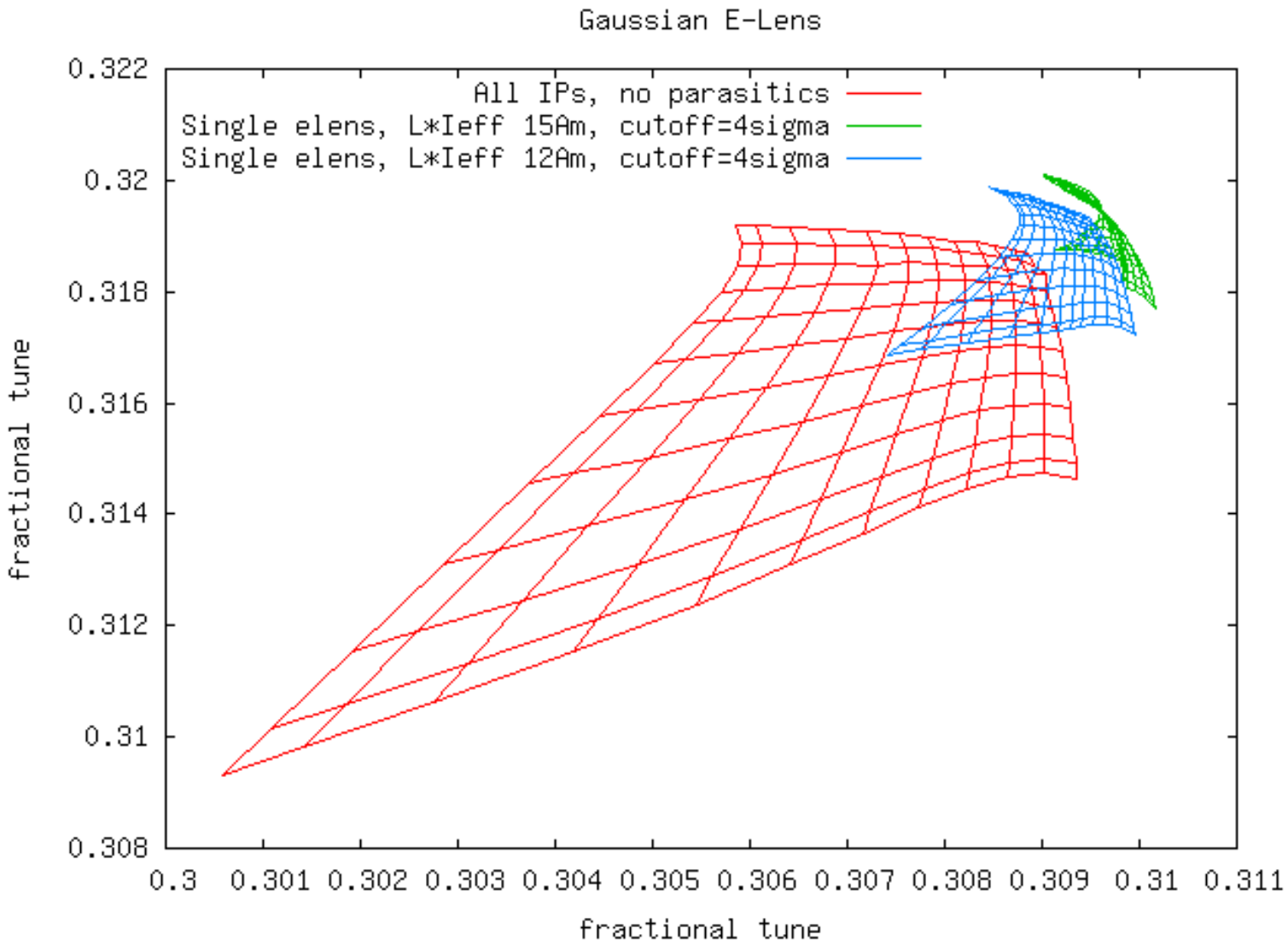
Has been implemented in `plibb` as a generalized current element. Different $I(r)$ distributions available:

- Cut-off Gaussian
- Fermi-Dirac
- Flat top (= Wire, using correct form factor)

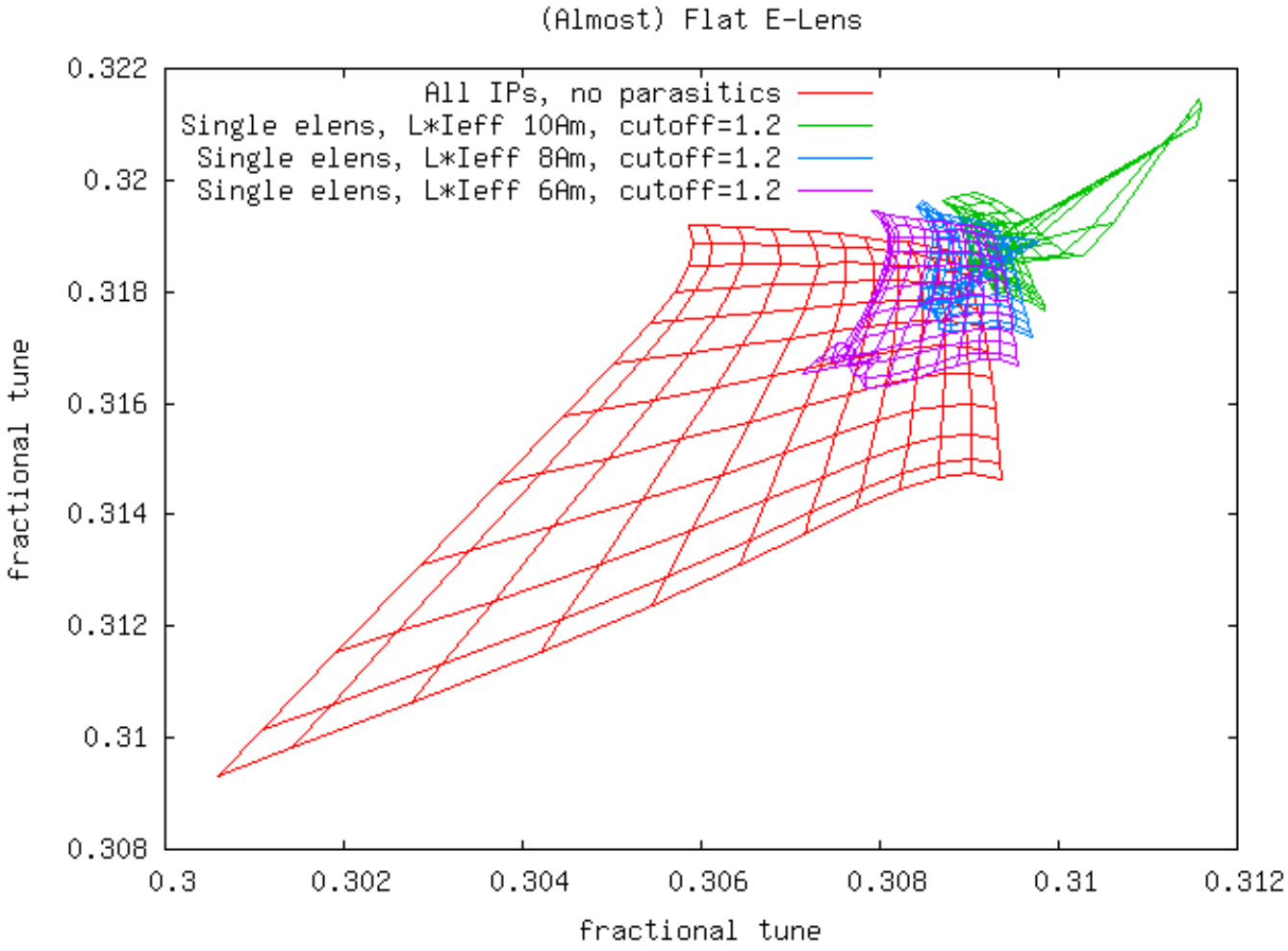
BB tune footprints



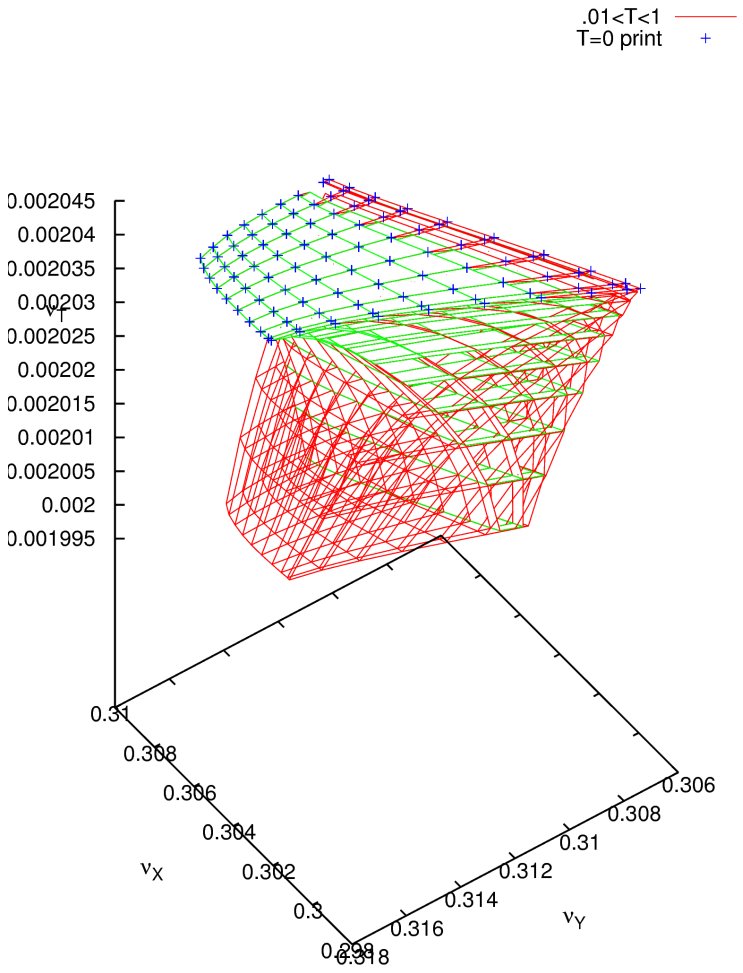
BB tune footprints w/Gaussian Electron Lens Element



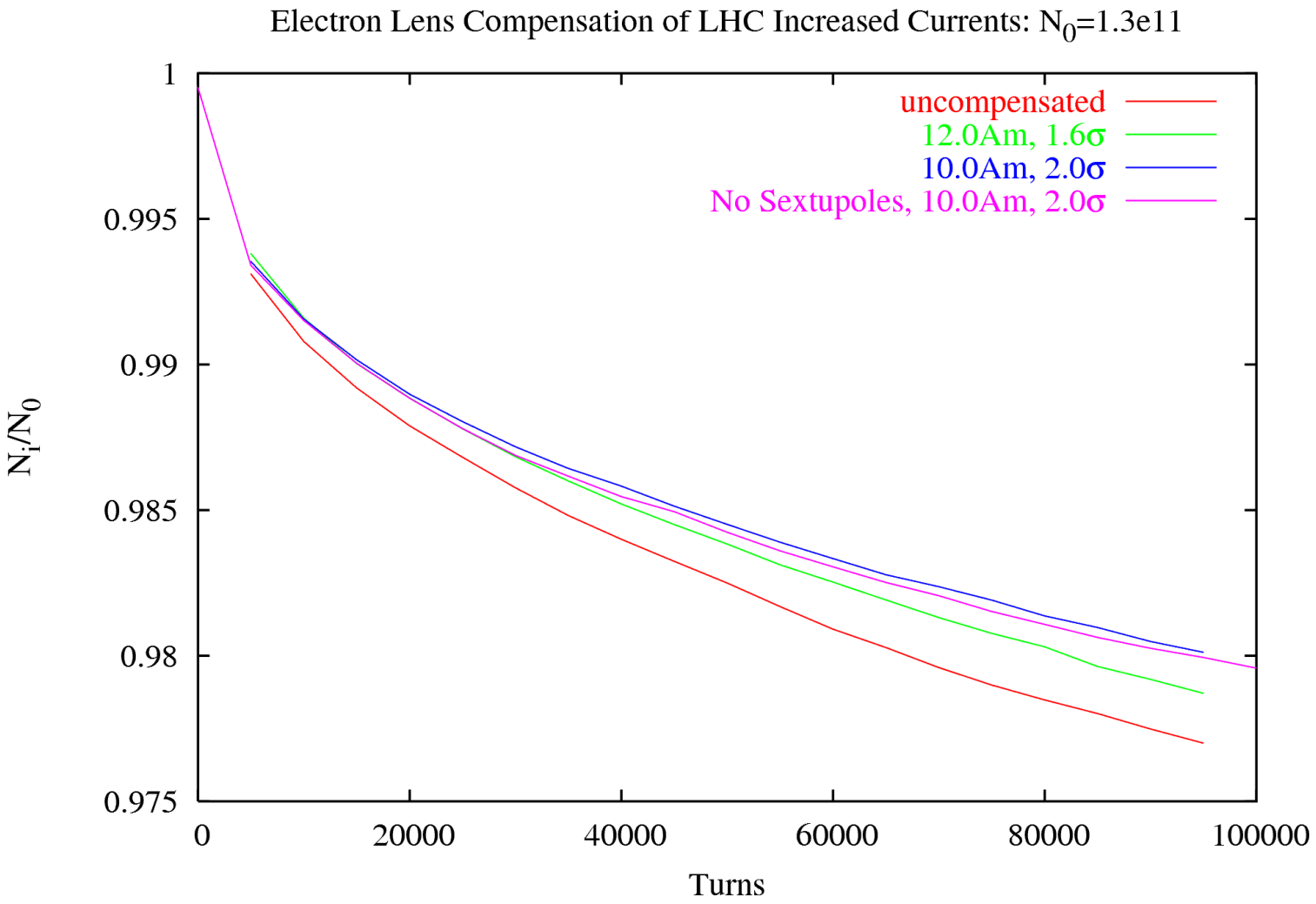
BB tune footprints w/quasi-flat Electron Lens Element



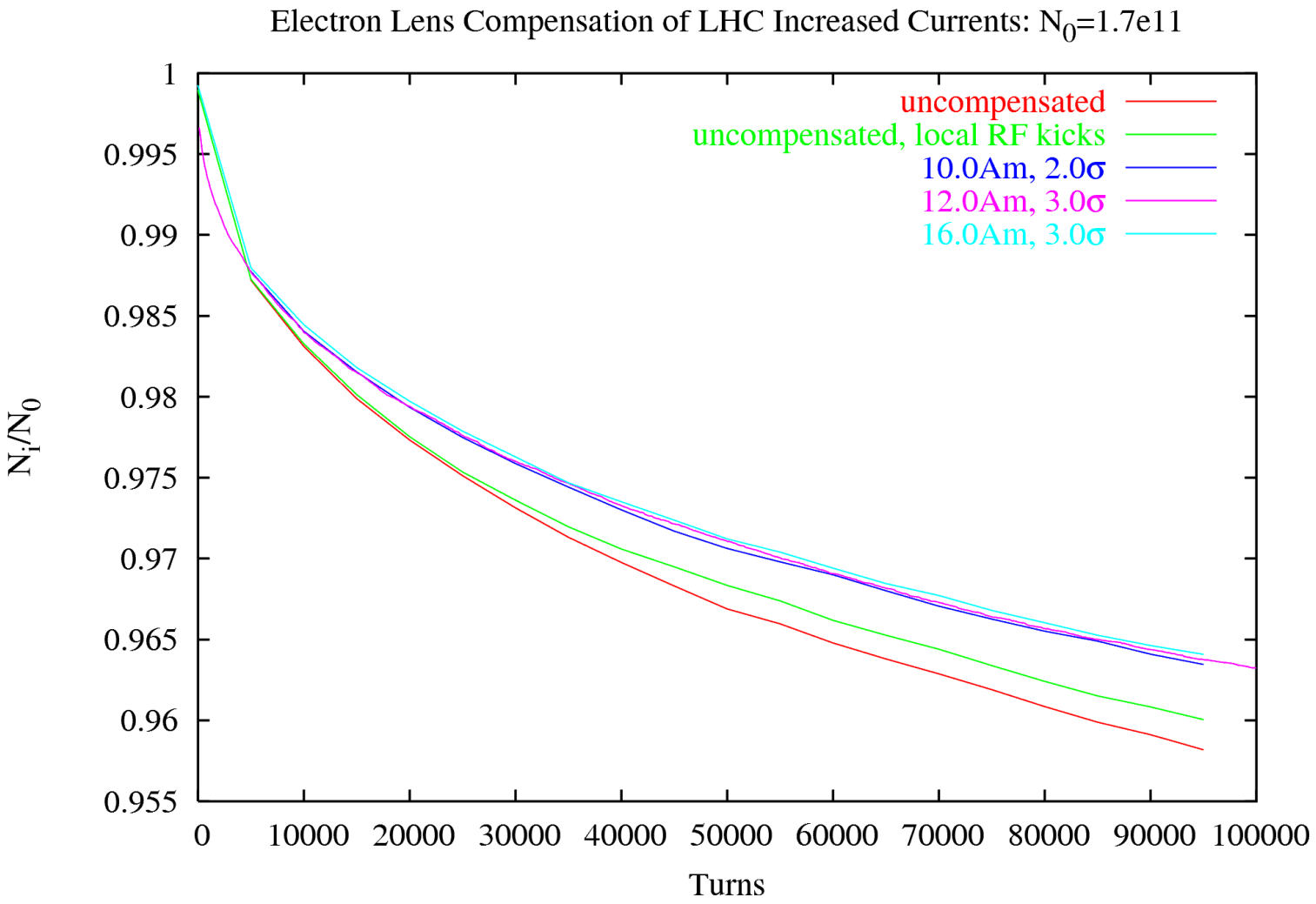
3D tune 'footprint'



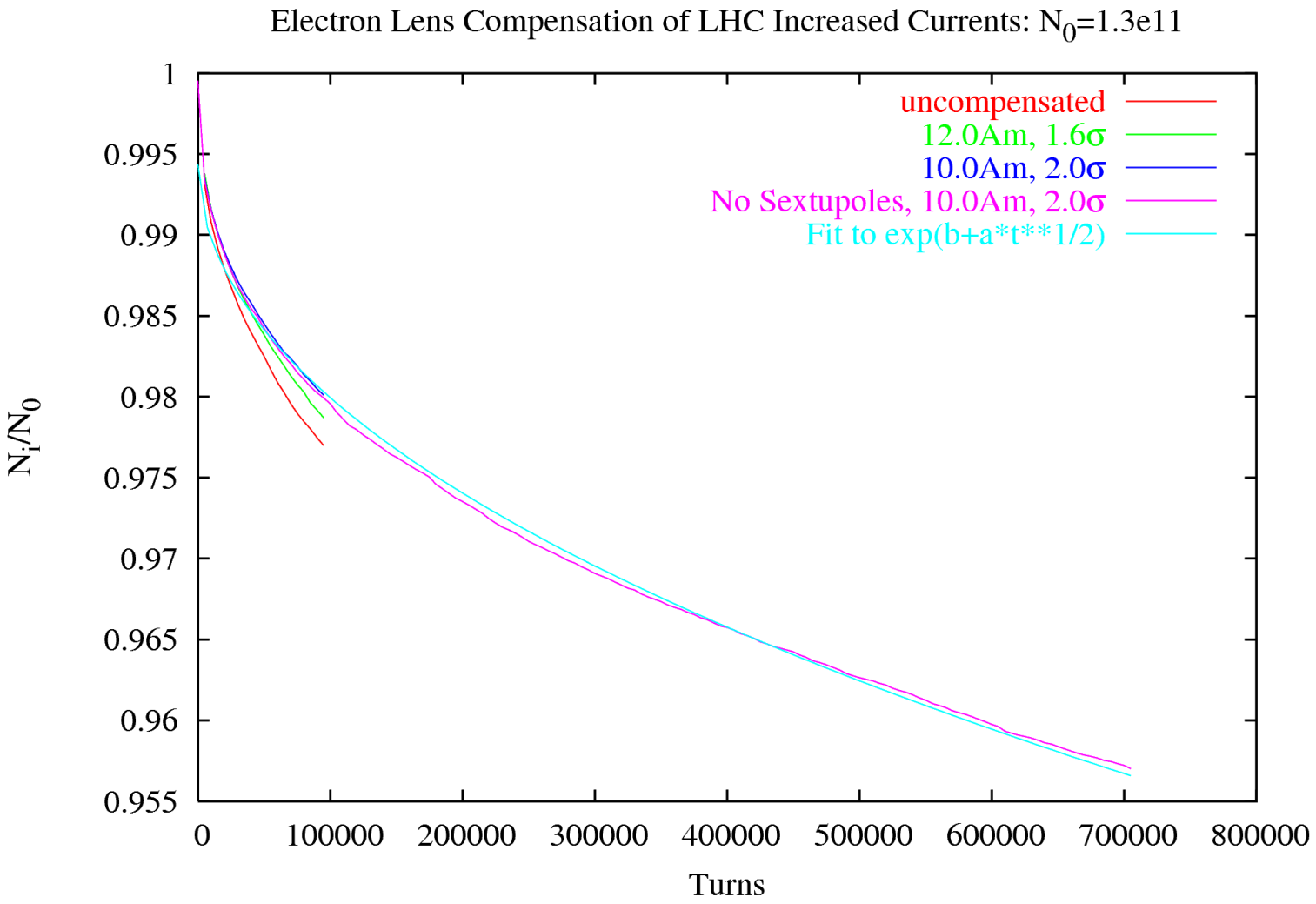
Electron Lens Tracking Results (I)



Electron Lens Tracking Results (II)



Electron Lens Tracking Results (III)



Label	$N_0 \cdot 10^{-11}$	$\int I_{eff} \mathrm{d}l$ [Am]	r_{max}/σ	τ	$(\Delta \log N)$
job1	1.3	0	n/a	$5h19m \pm 4m$	.24%
job5	1.3	-12	1.6	$5h14m \pm 5m$	.22%
job6	1.3	-10	2.0	$7h53m \pm 9m$	.29%
job7[a]	1.3	-10	2.0	$11h53m \pm 7m$	.56%
job2	1.7	0	n/a	$1h37m \pm 3m$	.53%
job0[b]	1.7	0	n/a	$1h40m \pm 6m$	0.43%
job3	1.7	-10	2.0	$2h20m \pm 5m$	0.61%
job4	1.7	-10	3.0	$2h18m \pm 1m$	0.57%
job9	1.7	-16	2.0	$2h02m \pm 1m$	0.41%

[a] No sextupoles, $7 \cdot 10^5$ turns
[b] RF kicks treated locally

Conclusions + Future Work

- Timescales of minutes are accessible to simulations
- Codes show similar + plausible behavior for LHC e-lens
- Differences need to be explained: bugs, approximations, lattices?
- **Benchmarking is needed**
- RHIC electron lens may be an ideal study object

Backup + Orphaned Slides

Possible E-Lens Configuration

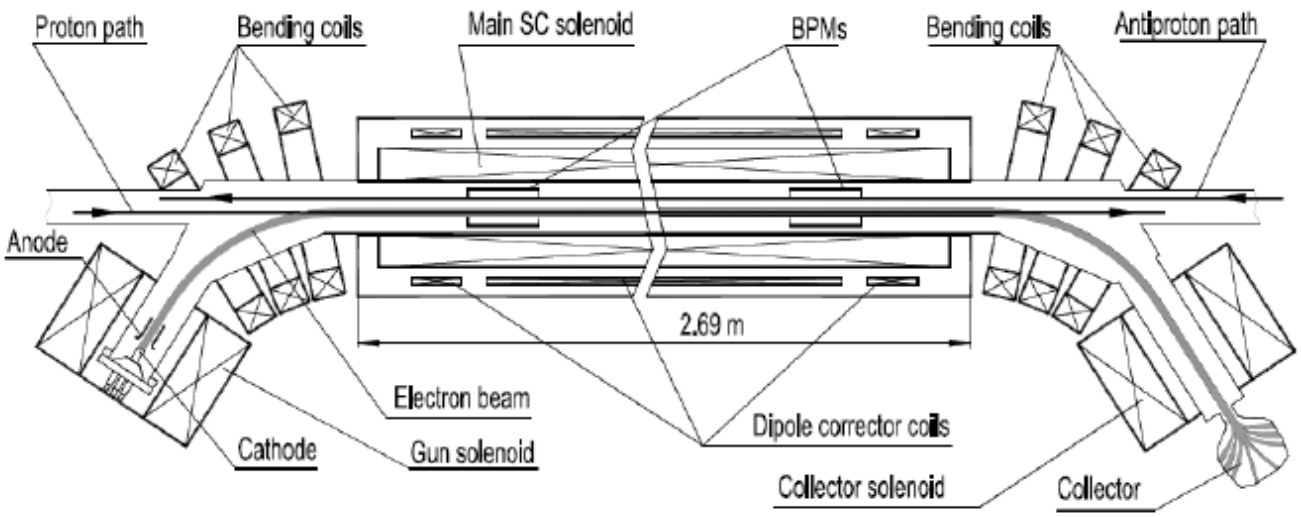
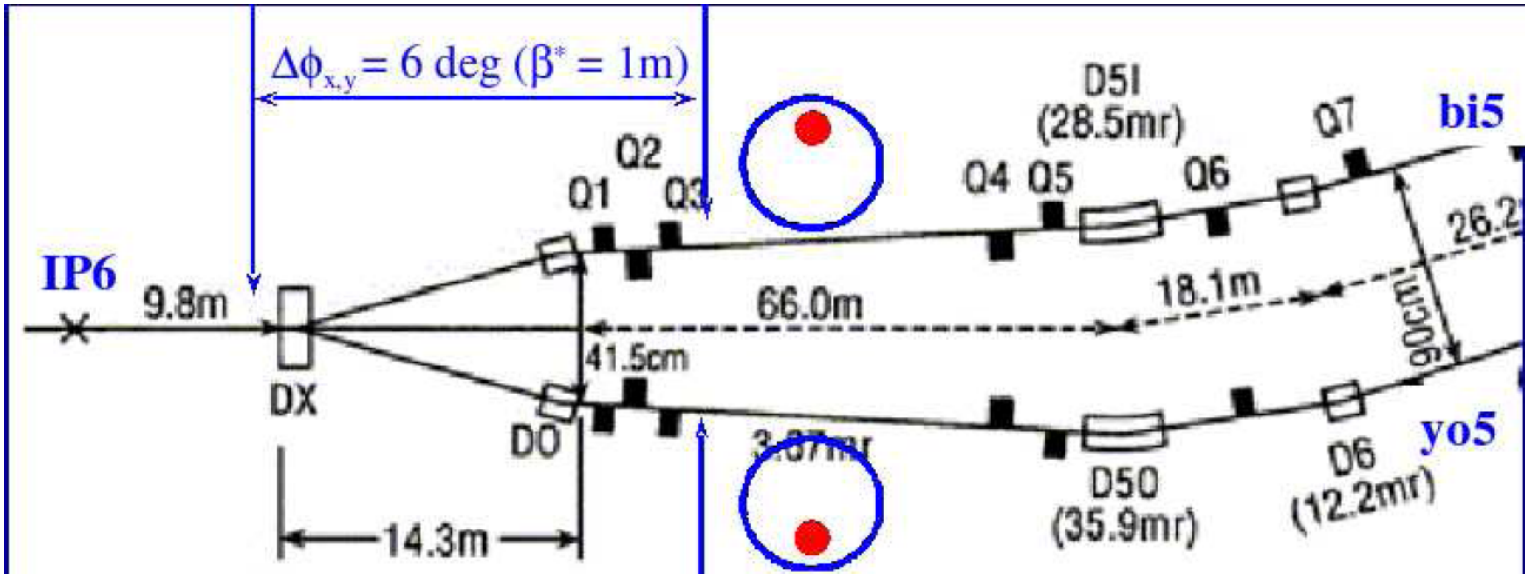
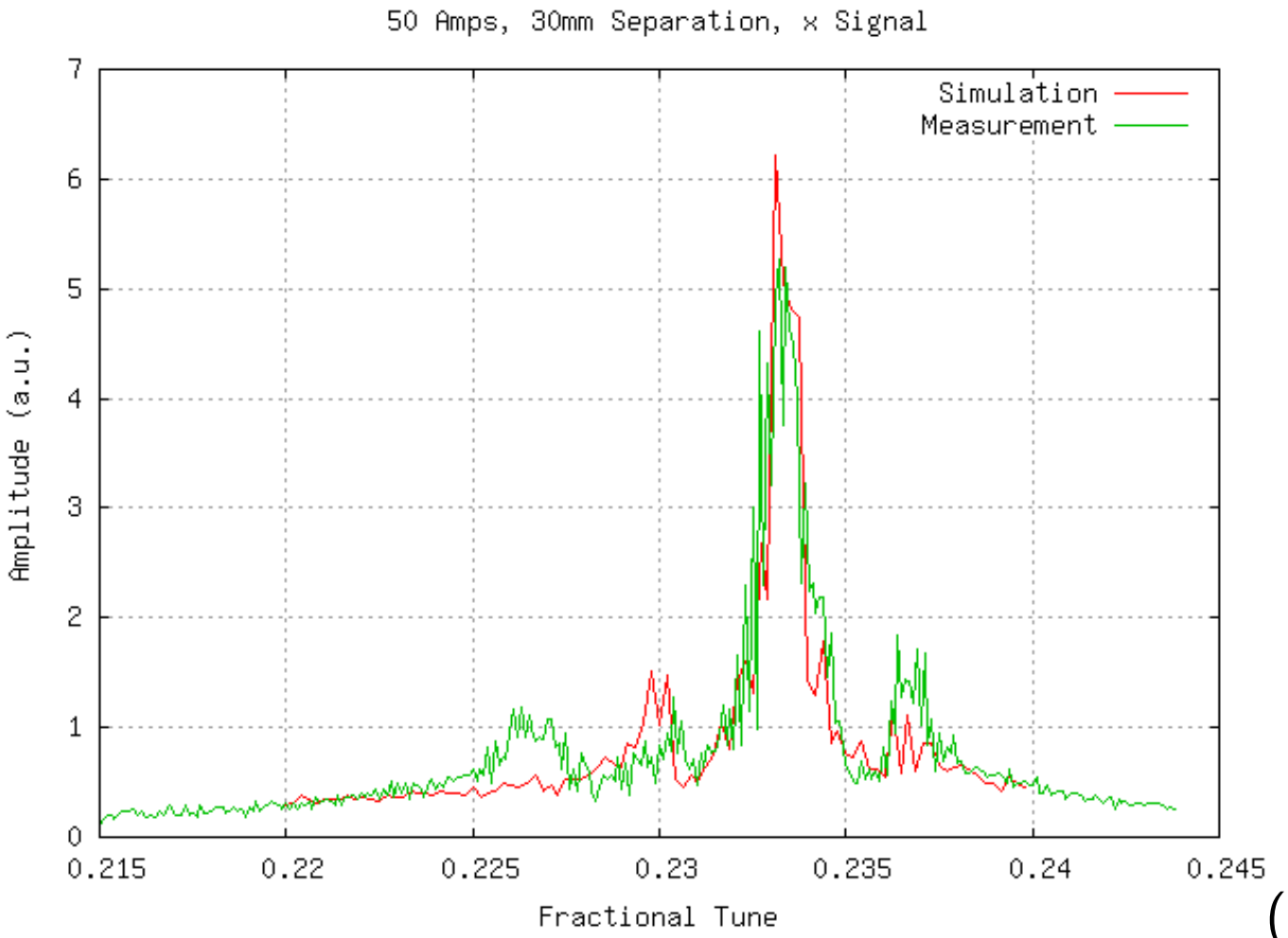


Figure 1: Electron lens configuration for compensation of beam-beam effects.

RHIC Wire Compensation

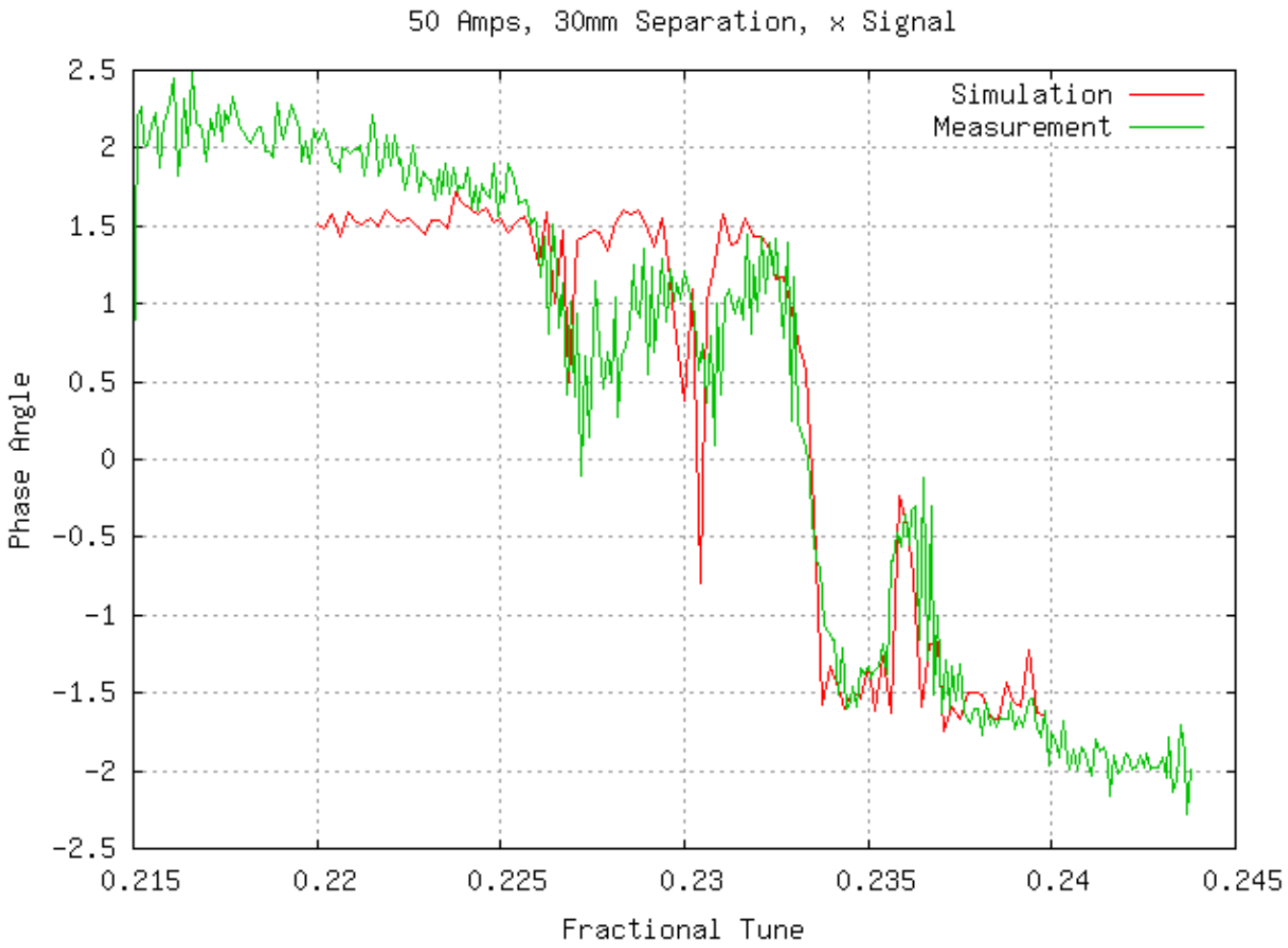
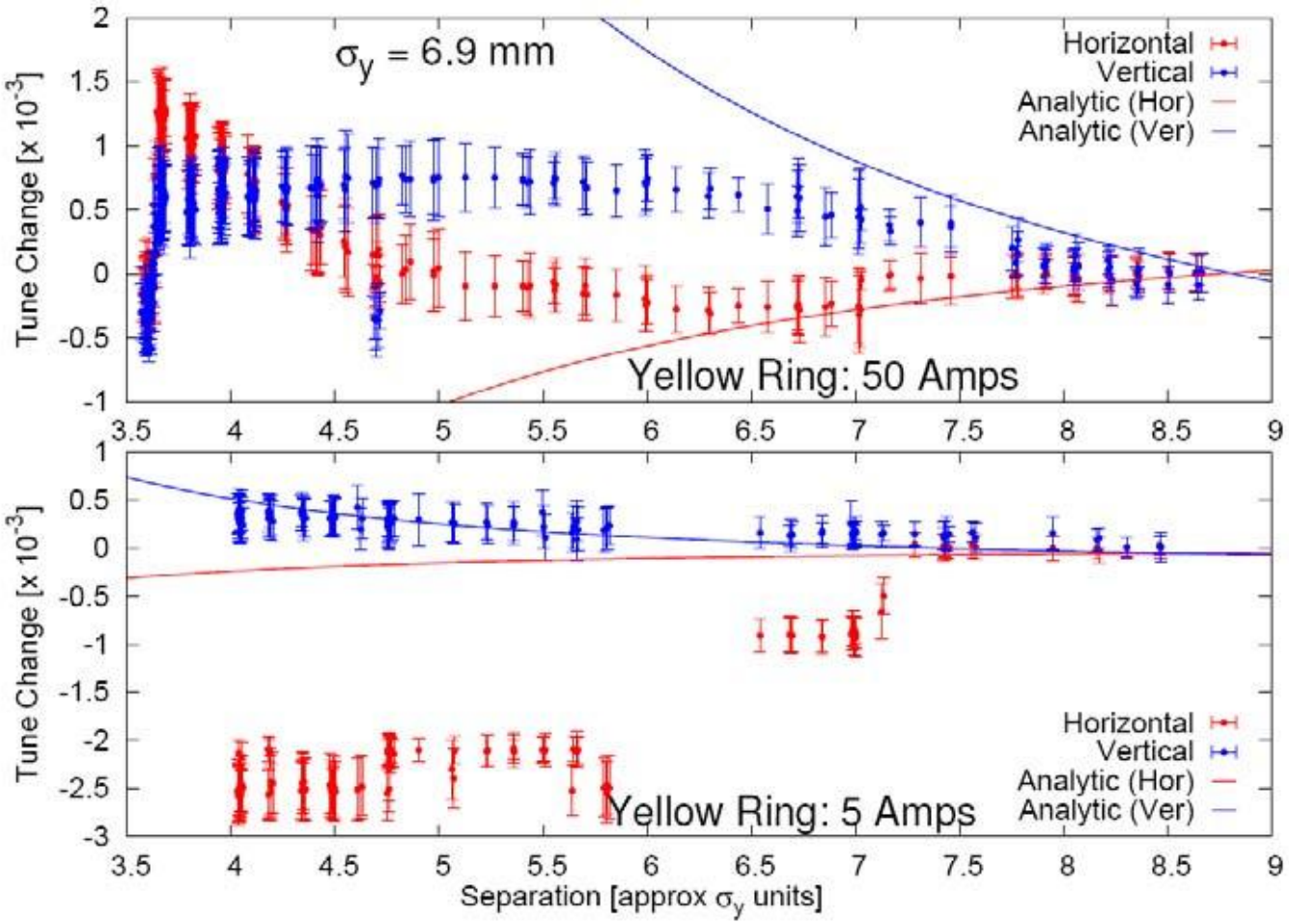


Past workshop; BTF Measurements



(BTF data courtesy W. Fischer, N. Abreu)

Past workshop; BTF Measurements



From N. Abreu's workshop presentation: