Modeling and Measurement of Amplitude Dependent Tune Shifts in CESR

Sarah Woodall

Lander University

David Rubin and Jim Shanks
Cornell University

Outline

- Background information about tune and origin of amplitude dependent tune shift
- Using a nonlinear oscillator model to become familiarized with Python and getting tune measurements from position data
- Simulations of tune shifts for varying amplitudes in different sextupole distributions
- CESR machine studies data and comparison with simulation

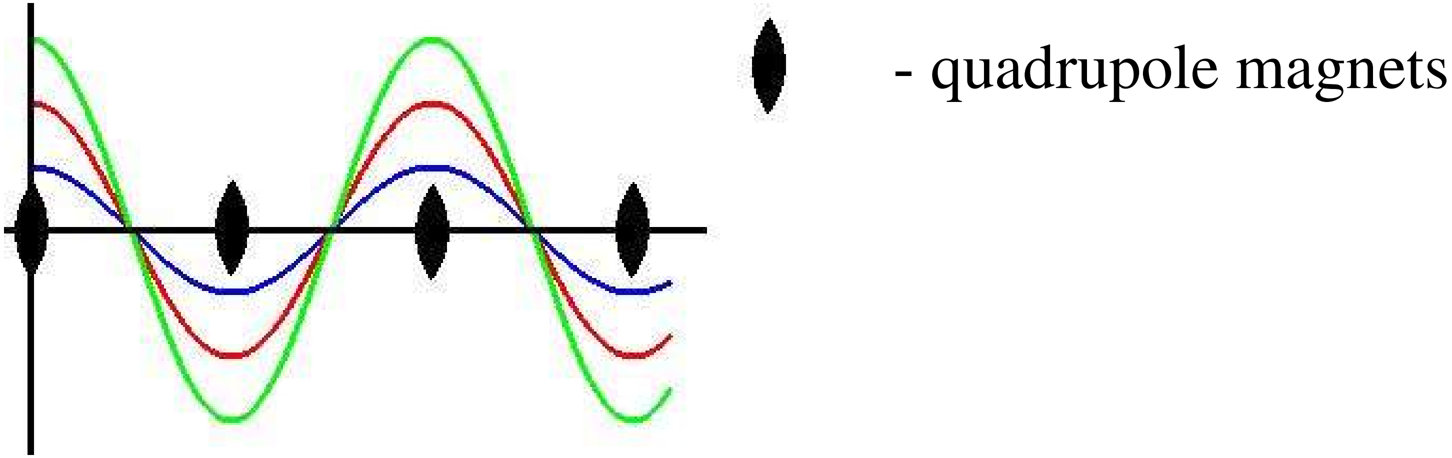
Oscillations in the Ring

- Quadrupole magnets, which are used for beam focusing, are responsible for the oscillation of the beam
  - F quadrupoles are horizontally focusing but vertically defocusing
  - D quadrupoles are vertically focusing but horizontally defocusing
- The tune, Q, is the number of oscillations that the beam experiences about its central axis as it passes through the beam pipe in a single turn
- As quadrupole focusing strength increases, the tune increases

$$Q = \frac{1}{2\pi} \oint \frac{ds}{\beta(s)}$$

Amplitude Dependent Tune Shift

- The transverse kick from the quadrupole magnets is proportional to the displacement from the central axis
  - Leaving the tune independent of the amplitude
  - The beam acts as a simple harmonic oscillator

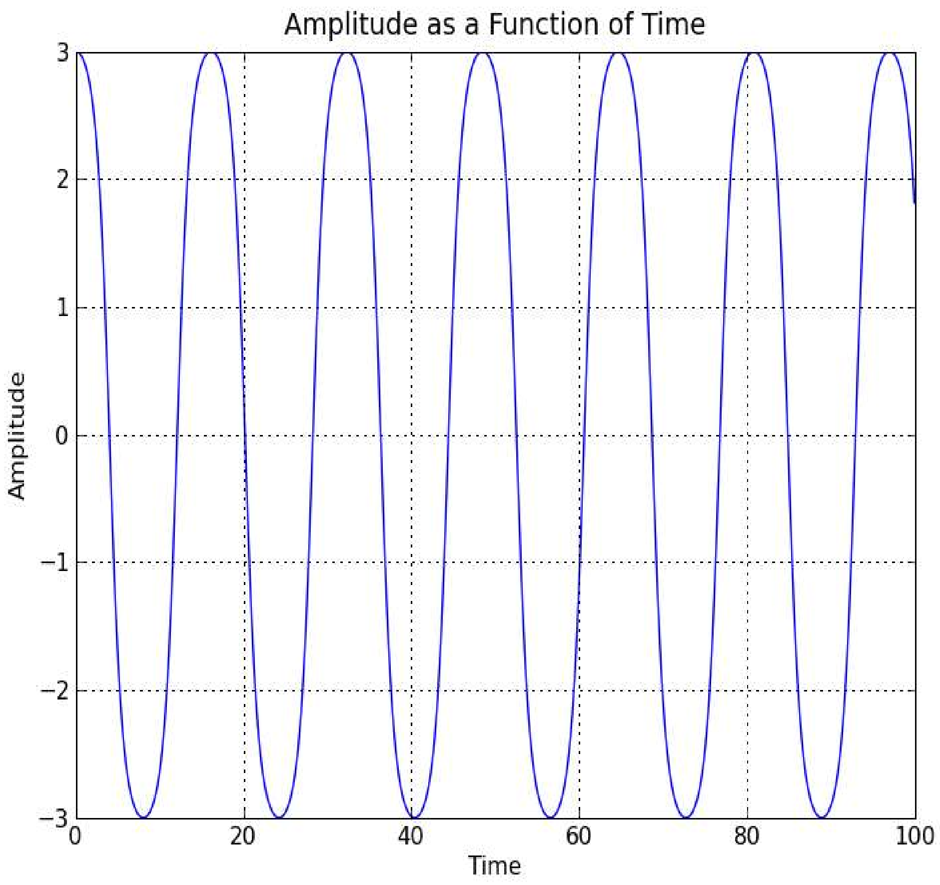


- Sextupole magnets are necessary for the correction of the beam energy spread
- The sextupole kick is nonlinear and as a result:
  - The tune becomes amplitude dependent

Nonlinear Oscillator Model

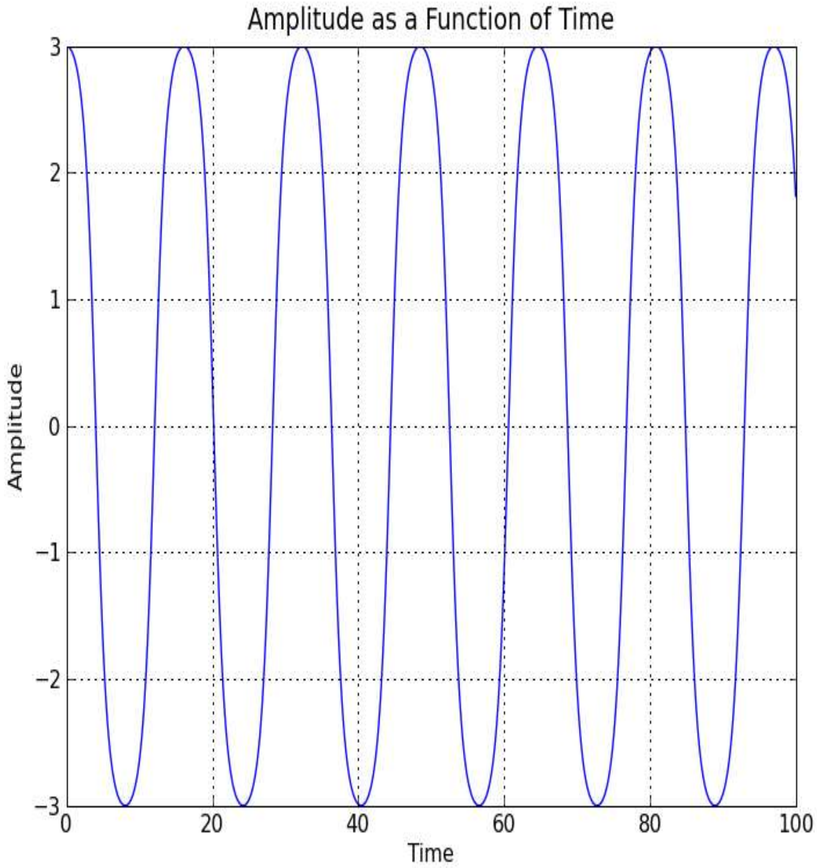
- Familiarizing myself with Python programming
  - Model of a nonlinear oscillator (pendulum)
  - Data from CESR is turn-by-turn data (position)

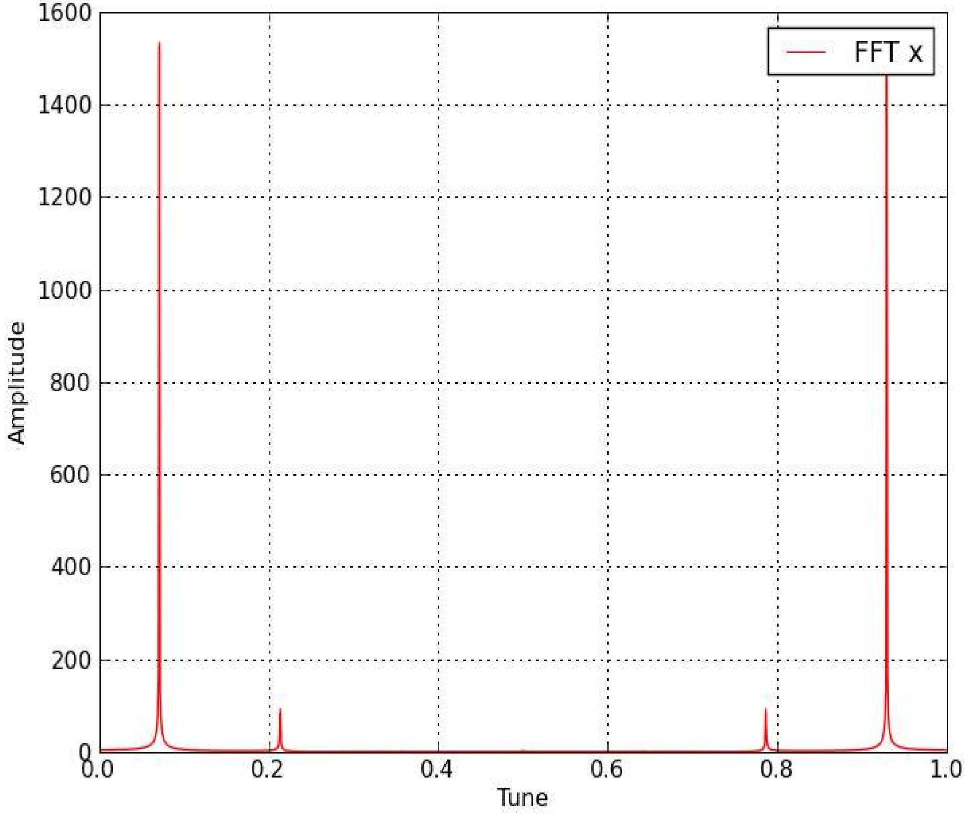
- Got position data by numerical integration of the differential equation
- With different initial amplitudes, found periods of each amplitude by locating local maxima
- Checked against the analytical solution for small angles using the small angle approximation



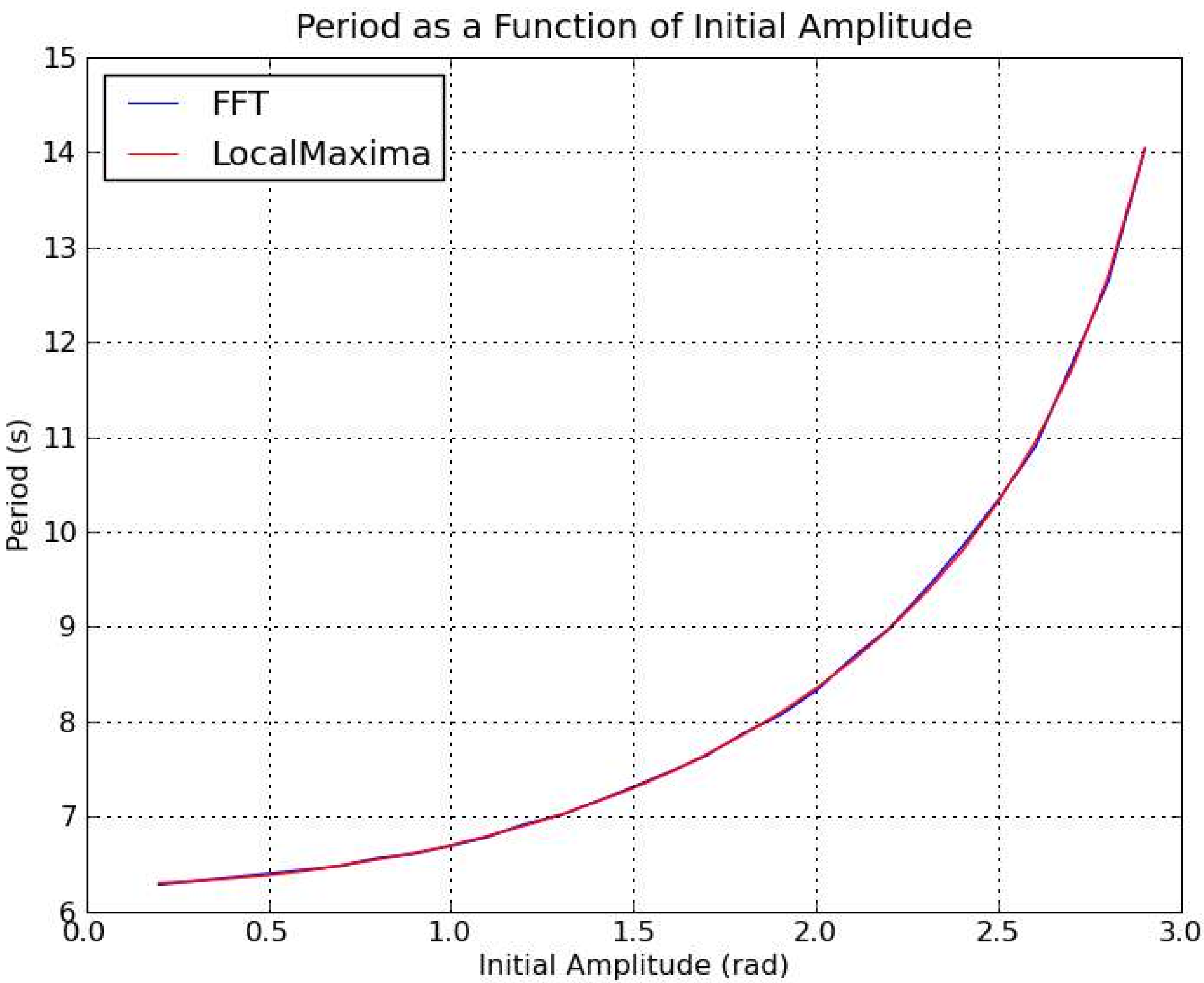
Nonlinear Oscillator Model

- Fast Fourier Transform
  - Takes position data and transforms into frequency data
  - Found periods by analyzing the frequency spectrum





Nonlinear Oscillator Model: Local Maxima Method v. FFT



Simulating Turn-by-Turn Data

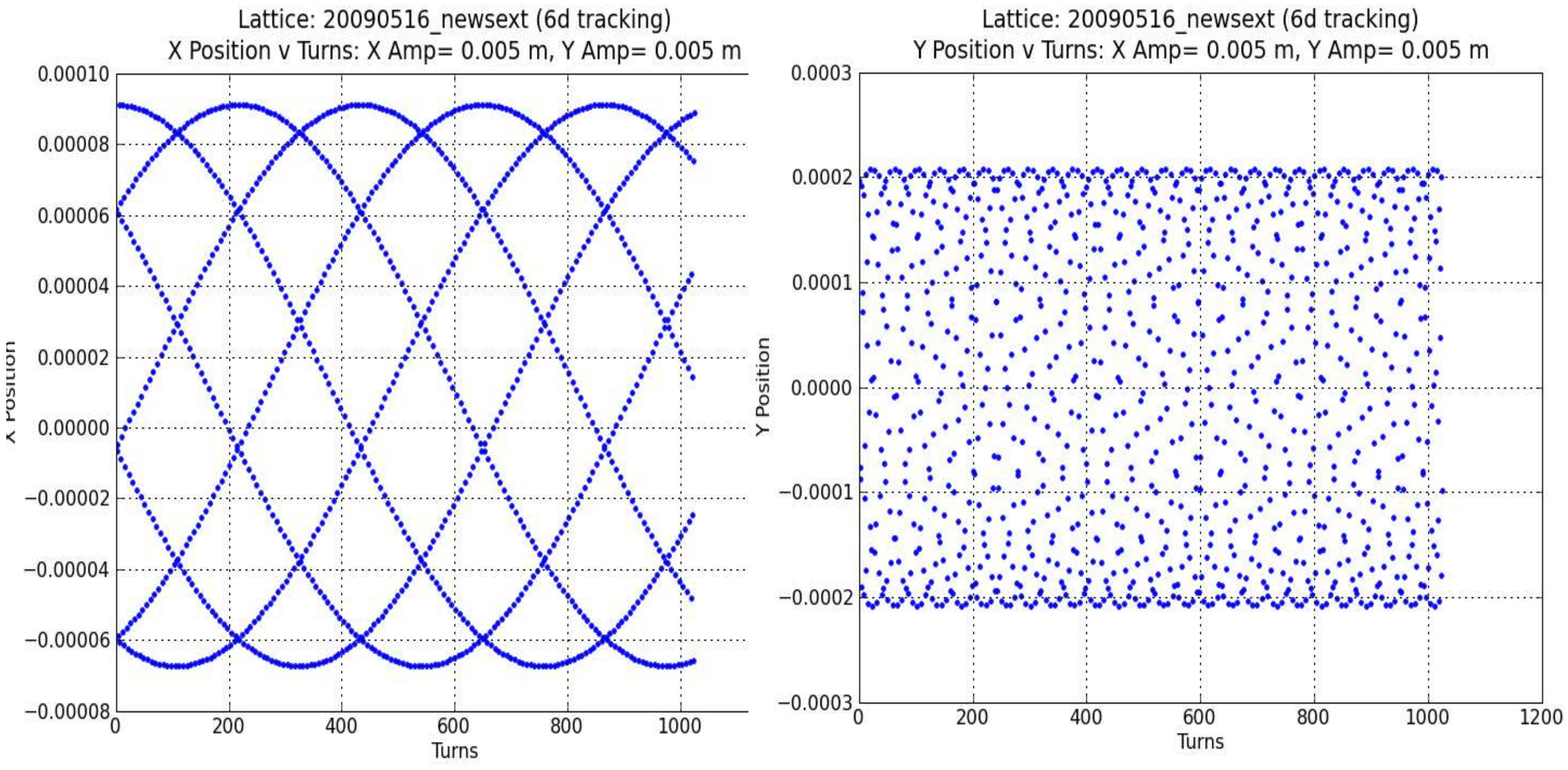
- Used Bmad subroutine library for simulations
- Simulation parameters
  - One bunch, 1024 turns
  - 4d/6d tracking
  - Four lattices, each with different sextupole distributions
  - Amplitudes ranging from .1 mm to 10 mm in x and y
  - Zero-amplitude tunes (Qx: .570, Qy: .629)
- As the beam passes a beam position monitor (BPM), values are returned for the position of the beam within the beam pipe, depending on how close the beam is to each of the four buttons that make up each BPM
- Data from the simulation is from a single BPM

Sextupole Distributions

- Two family sextupole distribution
  - Two families of sextupoles alternating throughout the ring: one positive, one negative
  - Specific values for the sextupole families are different and dependent on correcting off energy beam spread or chromaticity, while values within families are the same
- Optimized versions
  - These distributions correct for chromaticity and amplitude dependent tune shift and reduce various resonance-driving terms
  - Values for sextupoles are only symmetric between the east and west sides of the ring

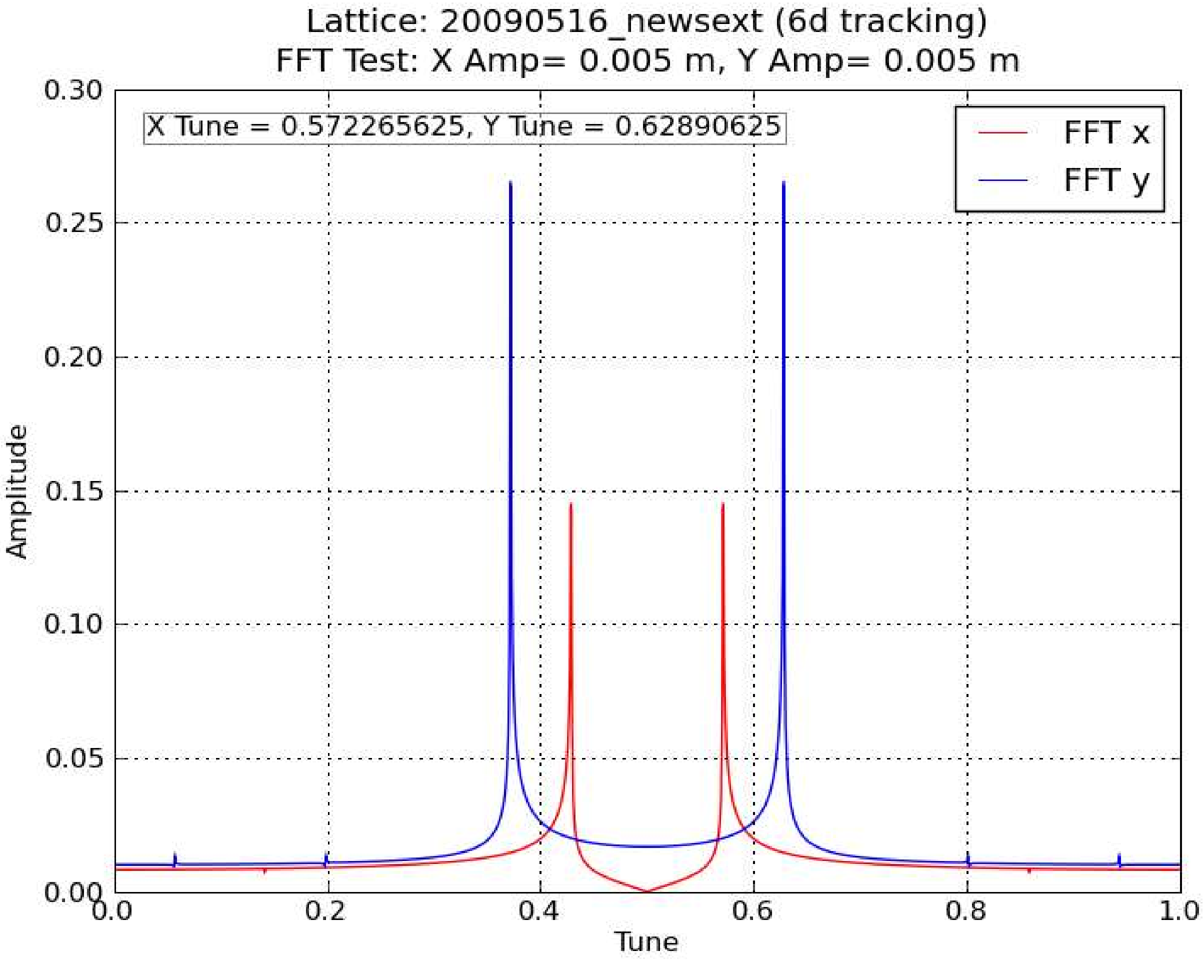
Finding Tune in Simulations

- For all amplitudes in each lattice, the horizontal and vertical tunes were found by taking an FFT of the position data



Calculating Tune Shift in Simulations

- Tune shift for each amplitude is calculated by subtracting the zero-amplitude tune from the tune at that amplitude: $\Delta Q_i = Q_i - Q_0$



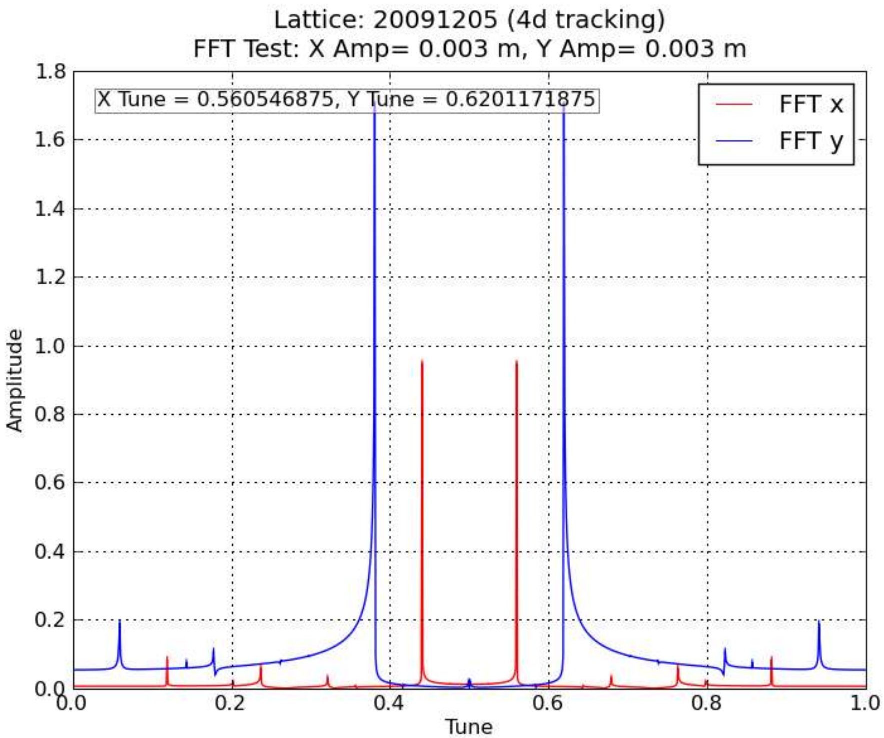
Determining Amplitude Dependence

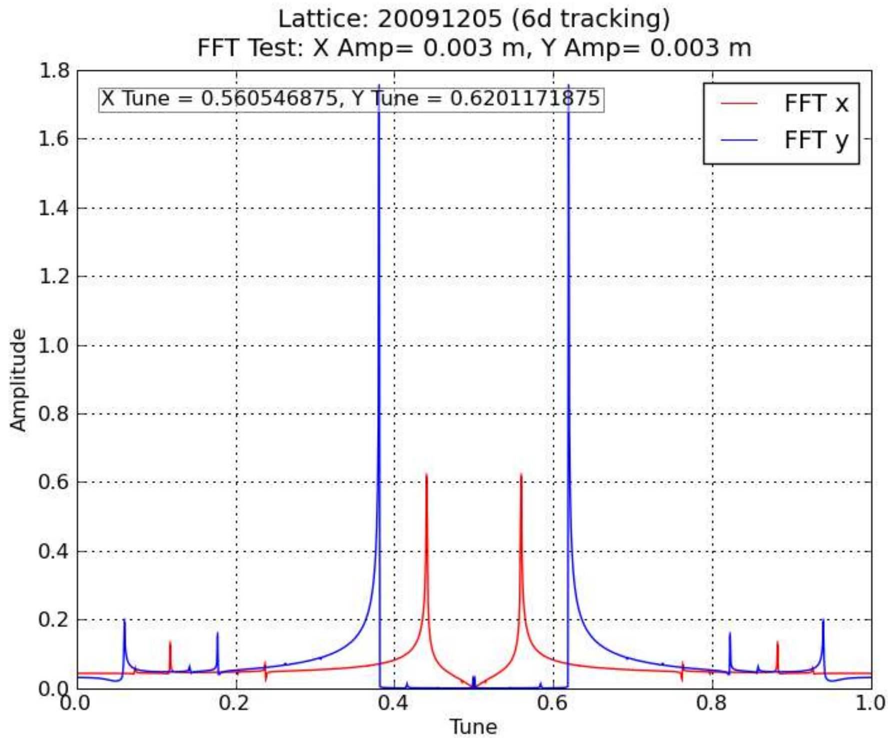
- For each sextupole distribution, the amplitude dependence of the tune shift was found by fitting a quadratic equation to:
    - X tune shift v. X amplitude
    - Y tune shift v. Y amplitude
- The quadratic coefficient is the term that shows amplitude dependence

Amplitude Dependence of Tune Shift				
Lattice Name (cta_2085mev)	20090516	20090516_newsext	xr20m_20091205	xr20m_20091205_newsext
Lattice Type	Two Family	Optimized	Two Family	Optimized
	6d Tracking			
Amp Dep of dQx	3.027E+02	2.729E+01	1.138E+02	2.203E+01
Amp Dep of dQy	3.461E+02	2.874E+02	2.760E+02	2.583E+02
	4d Tracking			
Amp Dep of dQx	3.113E+02	0.000E+00	1.237E+02	2.678E+01

Tracking Differences

- In comparing 4d and 6d tracking, the amplitude dependence of tune varies only slightly
- Although there are very few differences between 4d and 6d tracking, 6d was used to make the simulation as realistic as possible
- Synchrotron tune (.06) apparent in 6d tracking, not in 4d tracking

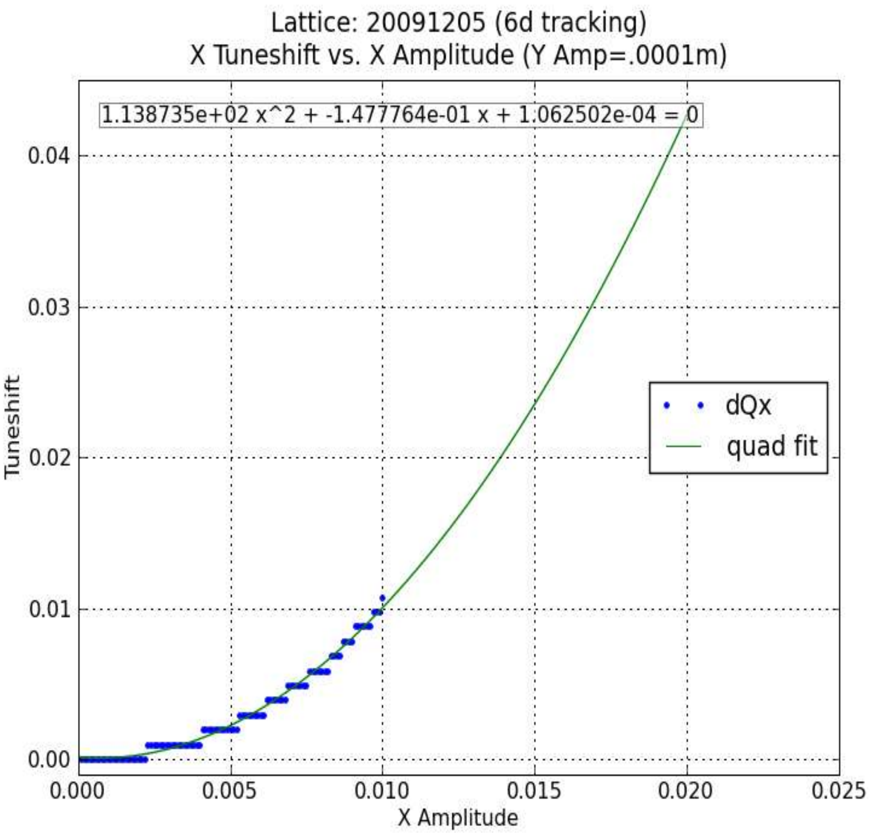




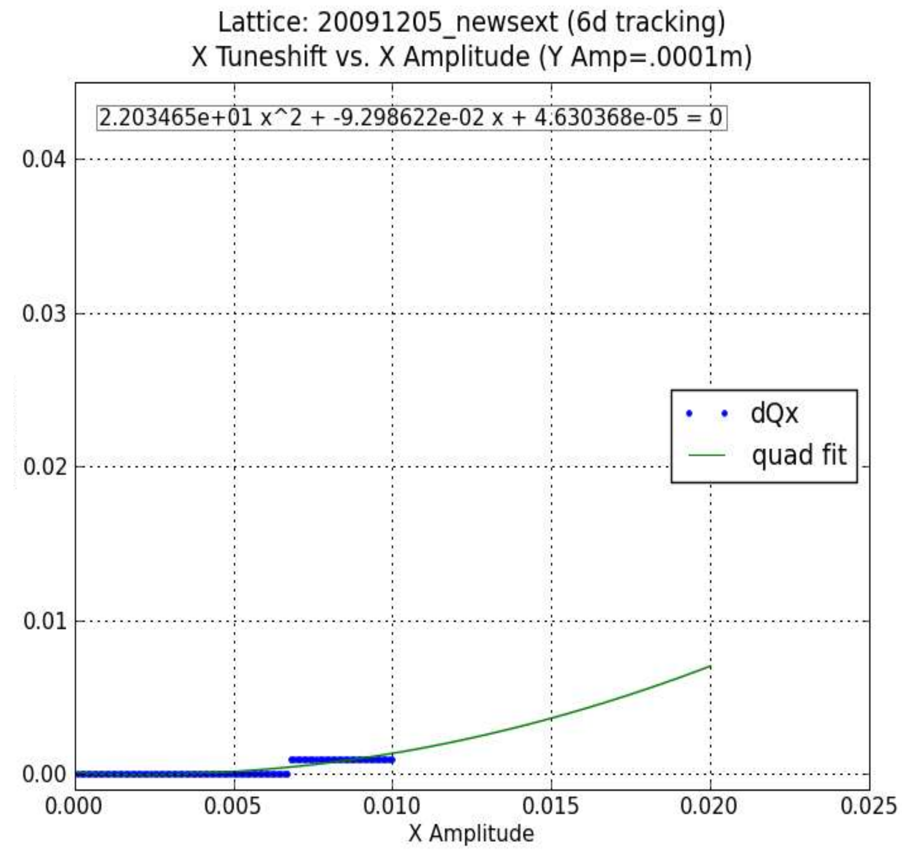
Sextupole Distributions

- X Tune Shift

Two Family



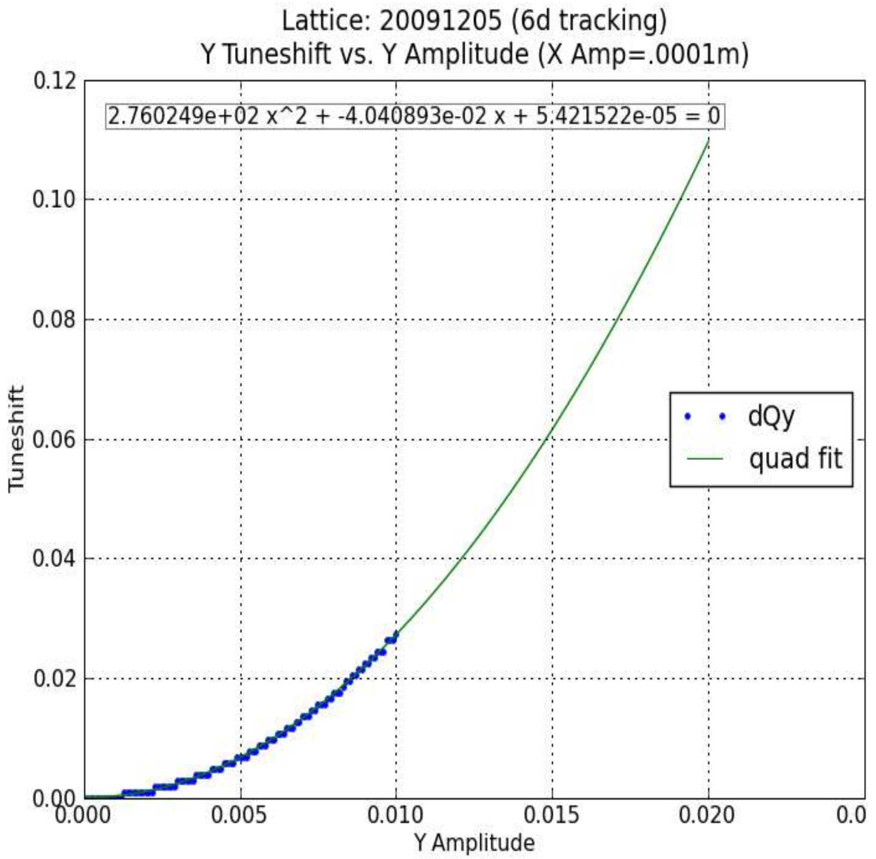
Optimized



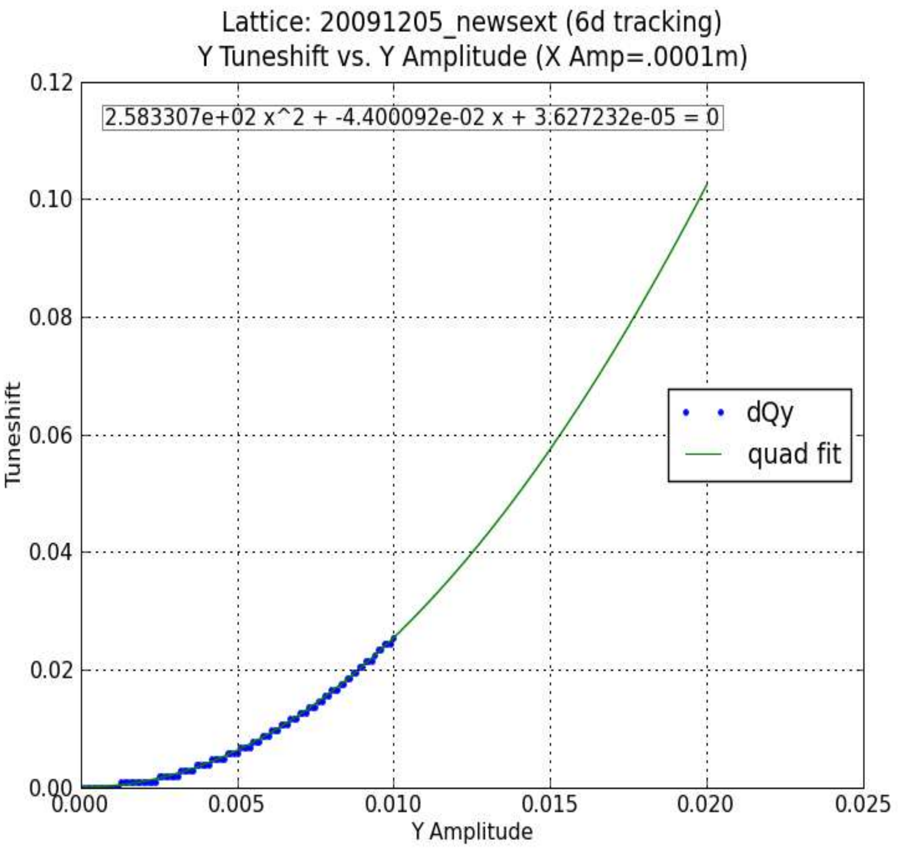
Sextupole Distributions

- Y Tune Shift

Two Family



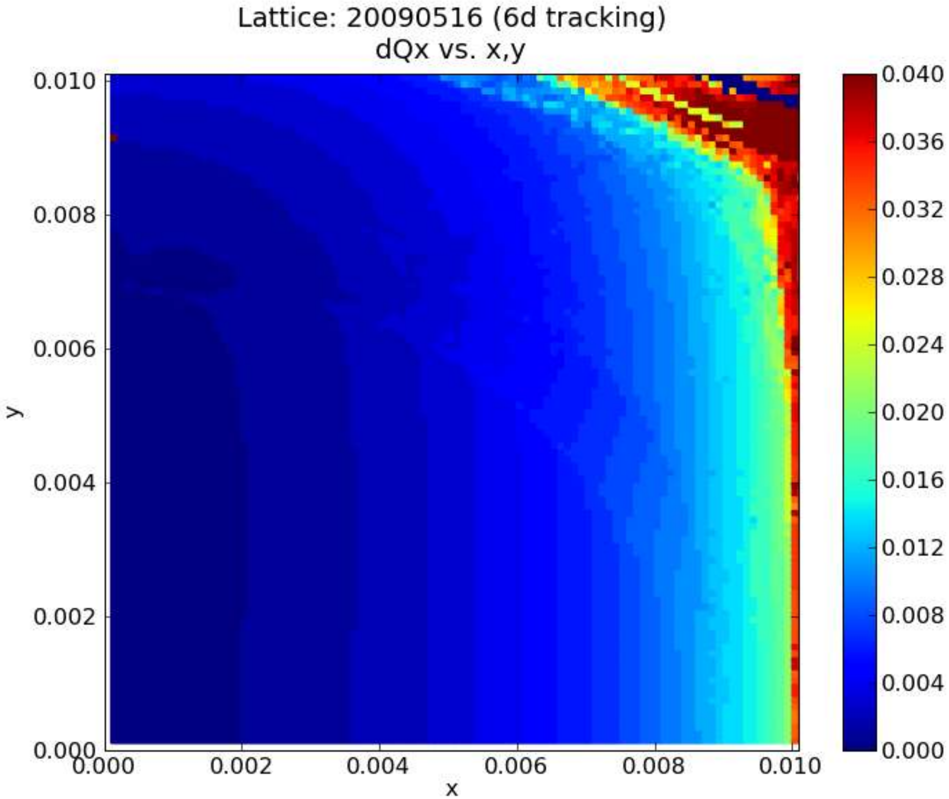
Optimized



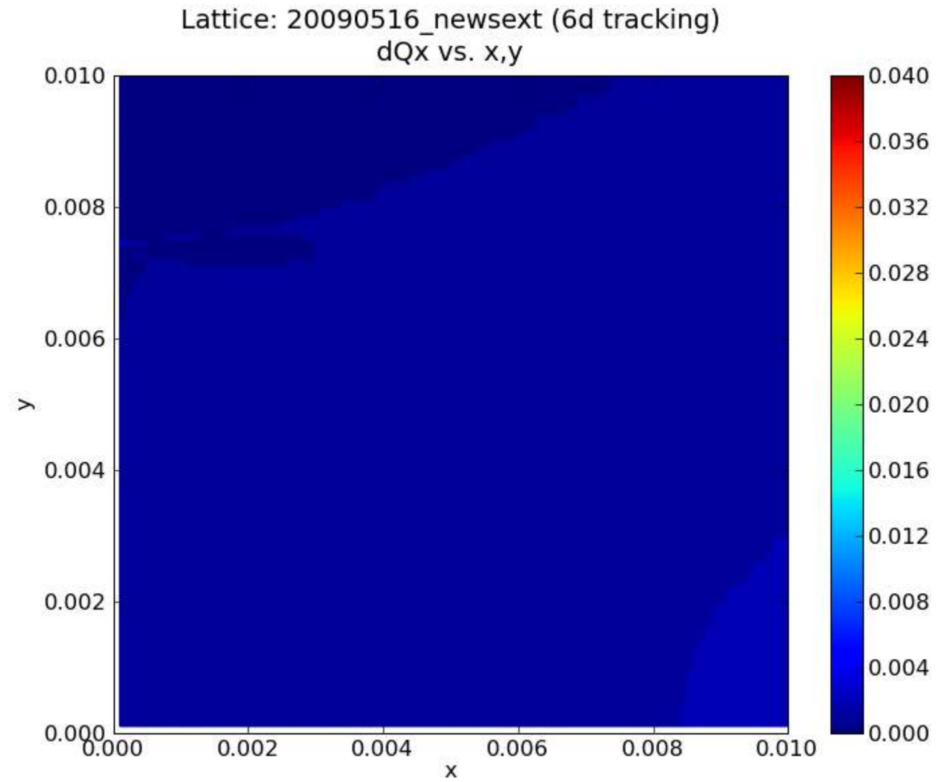
Sextupole Distributions

- X Tune Shift

Two Family



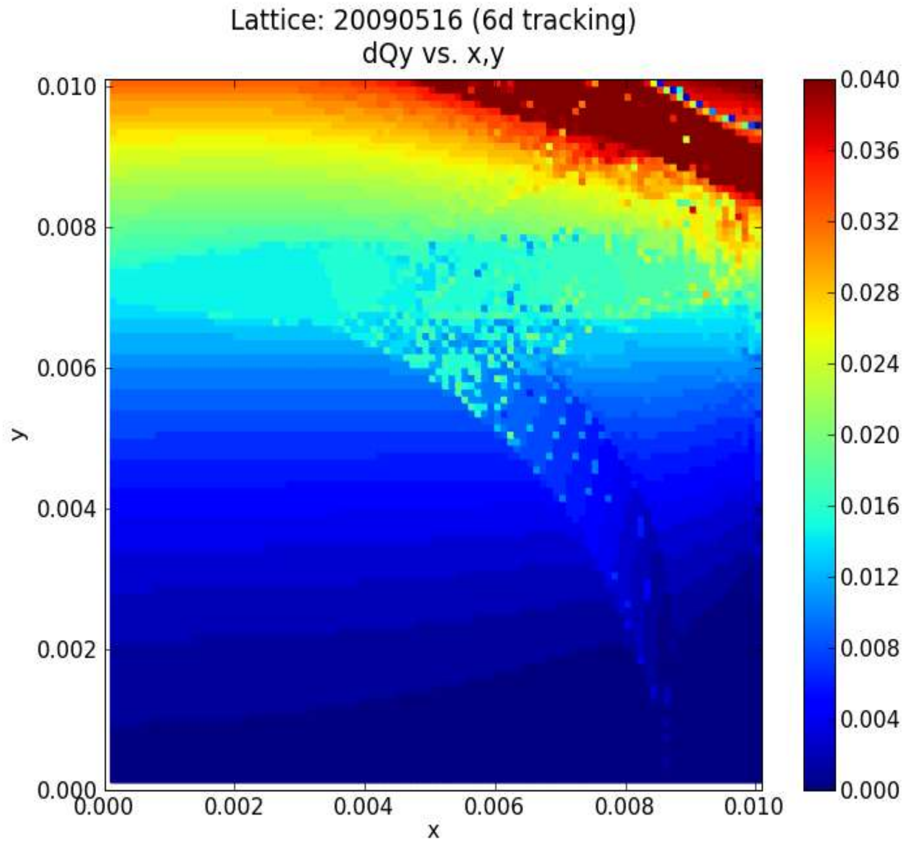
Optimized



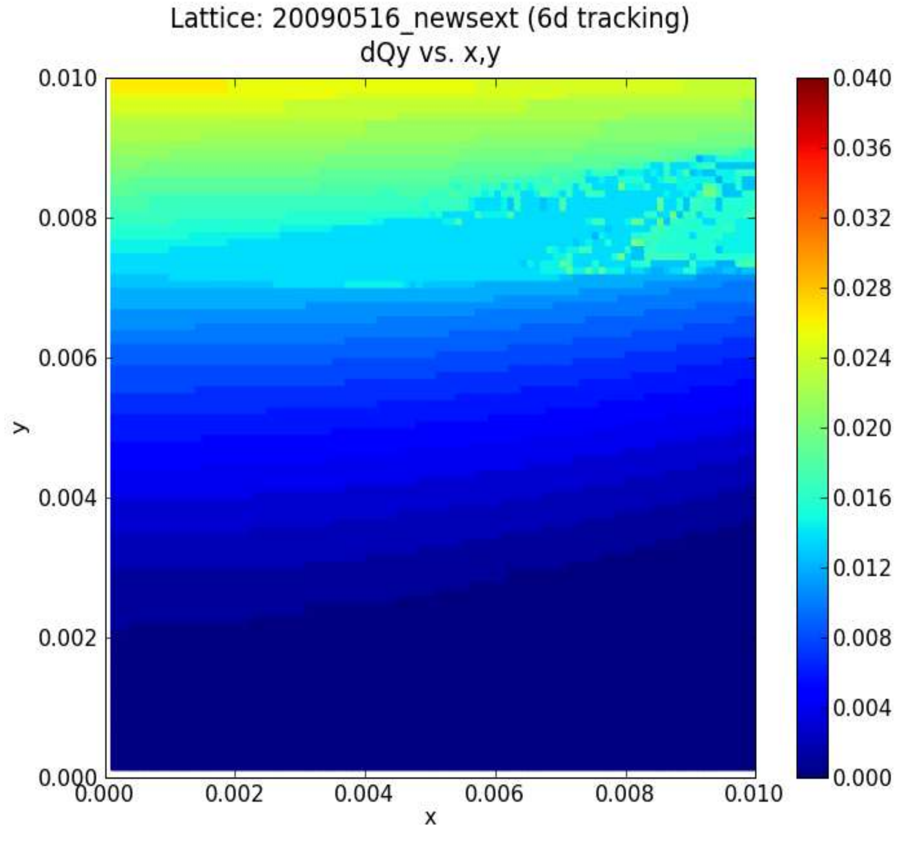
Sextupole Distributions

- Y Tune Shift

Two Family



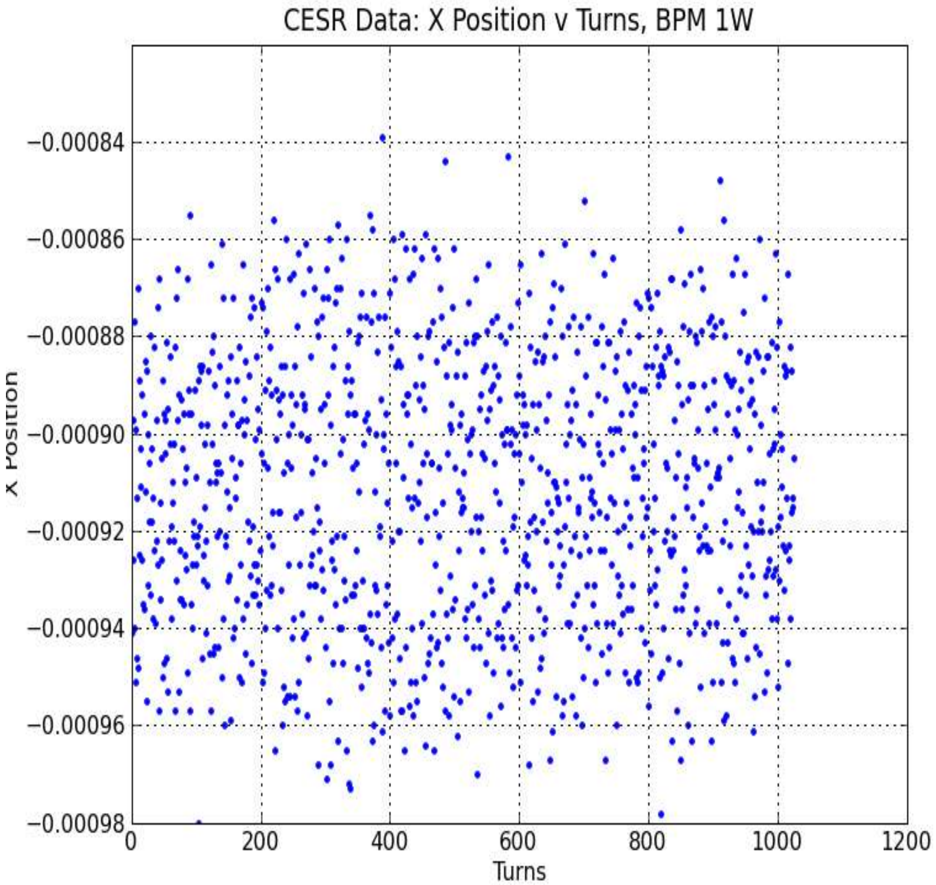
Optimized

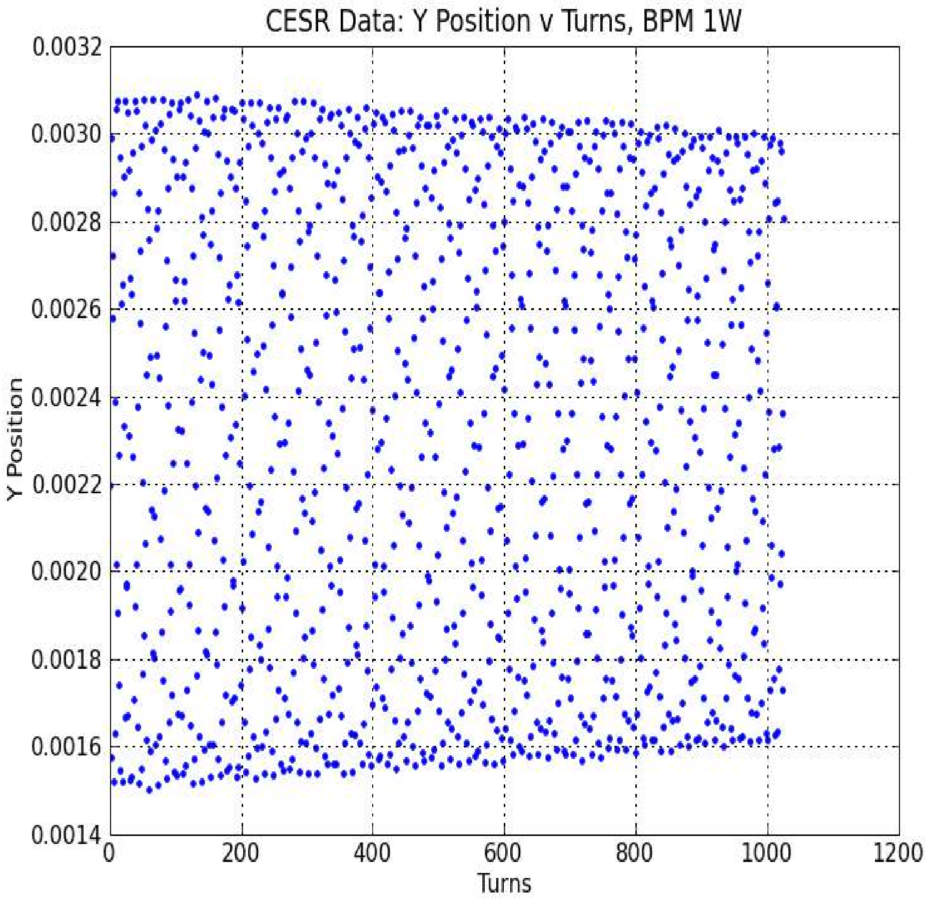


CESR Turn-by-Turn Data

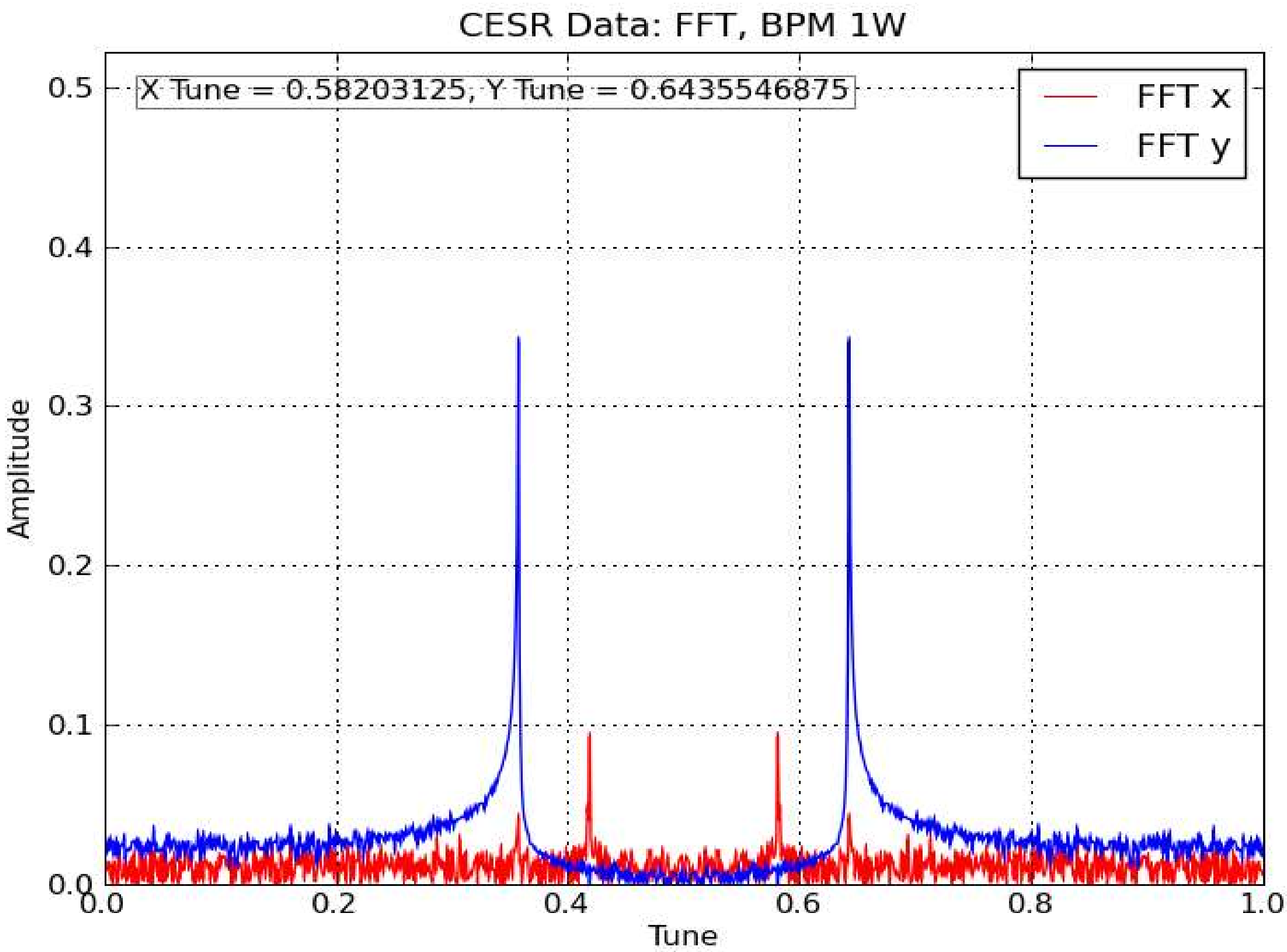
- Data acquisition parameters:
  - Single bunch, 1024 turns
  - Only used one sextupole distribution: cta_2085mev_xr20m_20091205 - two family distribution
  - Increased amplitudes by pinging the beam horizontally and vertically at separate times
  - Data collected for all BPMs
- Pinger - magnet turned on for a very short amount of time
  - Beam is kicked by pinger
  - TBT data collection is started right after the pinger kick
  - Greater pinger strength results in higher amplitudes

CESR Turn-by-Turn Position Data





CESR Turn-by-Turn Data: FFT



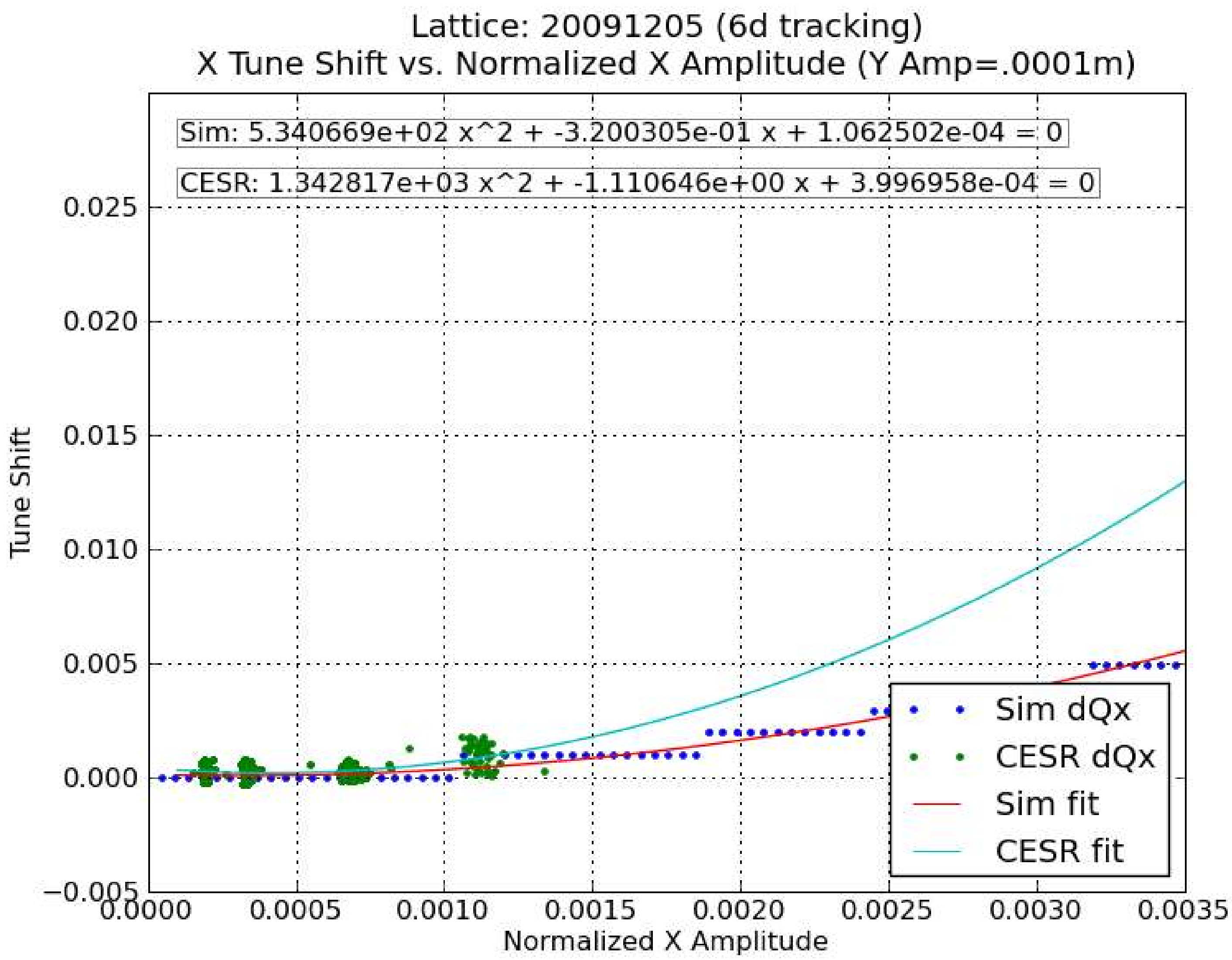
Comparing CESR Data to Simulations

- In order to compare the amplitude dependence of tune between CESR data and the simulations, the amplitudes needed to be normalized

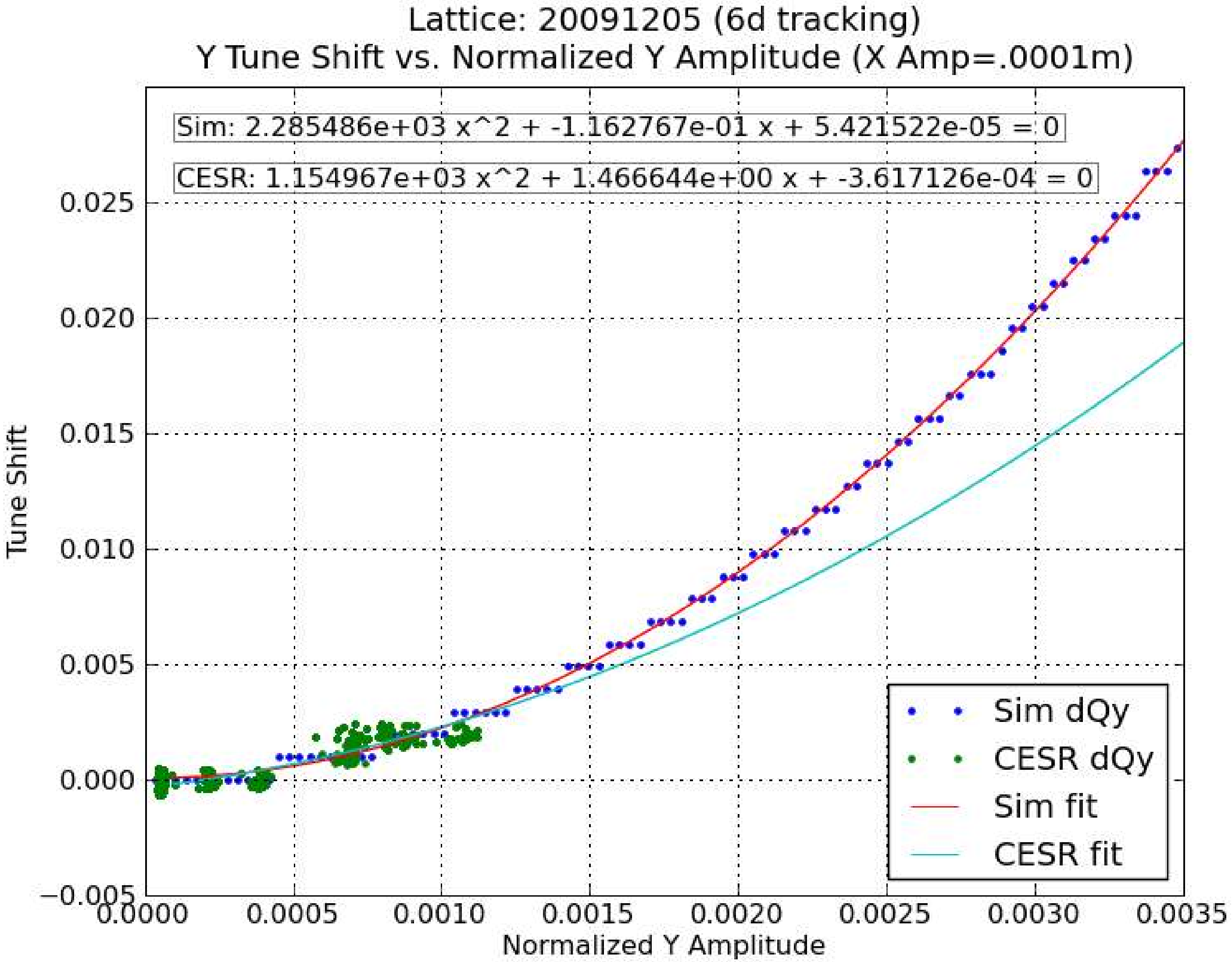
$$a = x\sqrt{\beta}$$

- For simulations, amplitudes were normalized using β at L0 (β_x: 4.69 m, β_y: 8.28 m) and the amplitudes that were used to create the data
- For CESR, TBT data was read in using Bmad, and the amplitude was normalized based on the amplitude and β at each specific BPM where data was taken

CESR v. Simulations: X Tune Shift



CESR v. Simulations: Y Tune Shift



Conclusions

- Measurements from CESR are not inconsistent with the model given the uncertainties of finding the amplitude dependence of the tune shift
- Based on the simulations, the optimized sextupole distributions might be further optimized to reduce the amplitude dependence of vertical tune shift
- **Future work:**
  - Larger amplitudes when measuring CESR data to have a better comparison with the model
  - Get data for more than one sextupole distribution when taking actual CESR measurements
  - Use interpolated FFT to find tunes for simulations to get better resolution
  - Understand the BPM systematics

In Summary

- Amplitude dependent tune shifts arise from nonlinearities in the ring caused by sextupole magnets
- For a variety of amplitudes, the tunes can be calculated by performing an FFT on position data
- $\Delta Q_i = Q_i - Q_0$
- Tune shifts can be compared between simulations and CESR by normalizing the amplitudes

Acknowledgments

- Thanks to:
  - My mentor, David Rubin
  - Graduate student, Jim Shanks
  - CLASSE
  - NSF