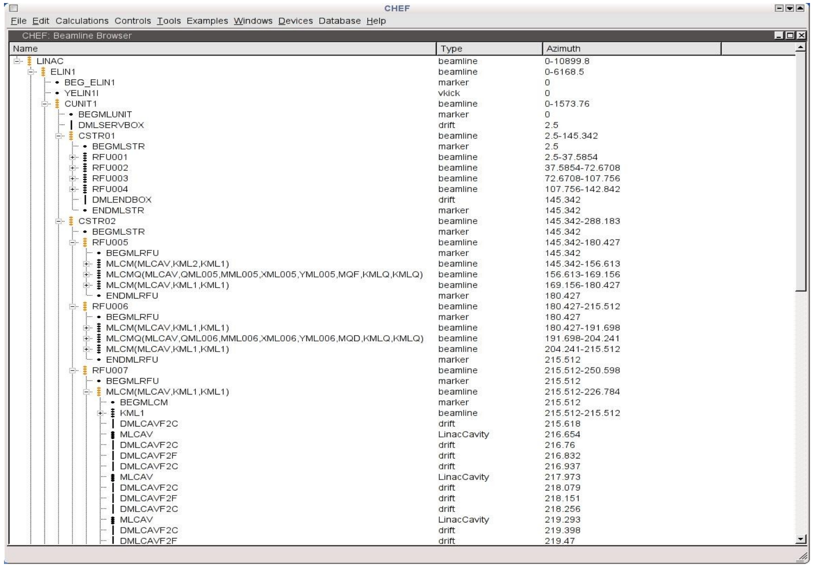
CHEF: Status Report

Jean-Francois Ostiguy
FNAL

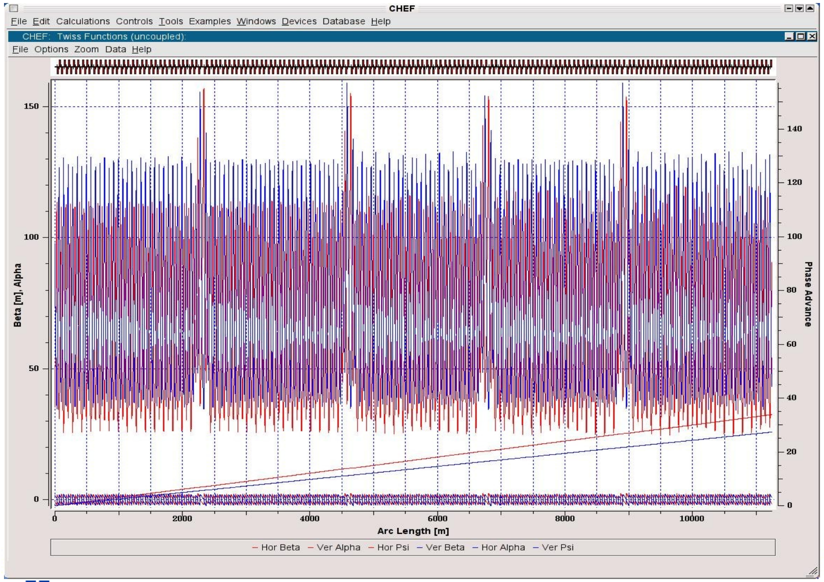
What is “CHEF” ?

- A framework for beam dynamics simulations consisting of a set of libaries organized hierarchically + python bindings + a standalone application
  - Written in std C++ (use STL and templates extensively)
  - Most functionality conveniently available through Python bindings.
- Originally designed for proton rings and beamlines; later, adapted for high energy linacs
  - The code provides facilities for both conventional tracking and map computions using automatic differentiation. The same generic code is used for both functions. Since most codes of this kind tend to be dominated by bookkeeping; the code design strives to make bookeeping as generic as possible to minimize need for “re-invention”. This makes it straighforward to accommodate special needs.
- Some Distinctive Features:
  - No a-priori, embedded paraxial approximation
  - Can accurately accommodate large dp/p
  - Consistently uses 6D canonical variables (i.e. px/p , not x')
  - No a-priori embedded relativistic (beta ~ 1) approximation
  - In principle, can track phase space “patches” using DA variables.

Some Features of Interest

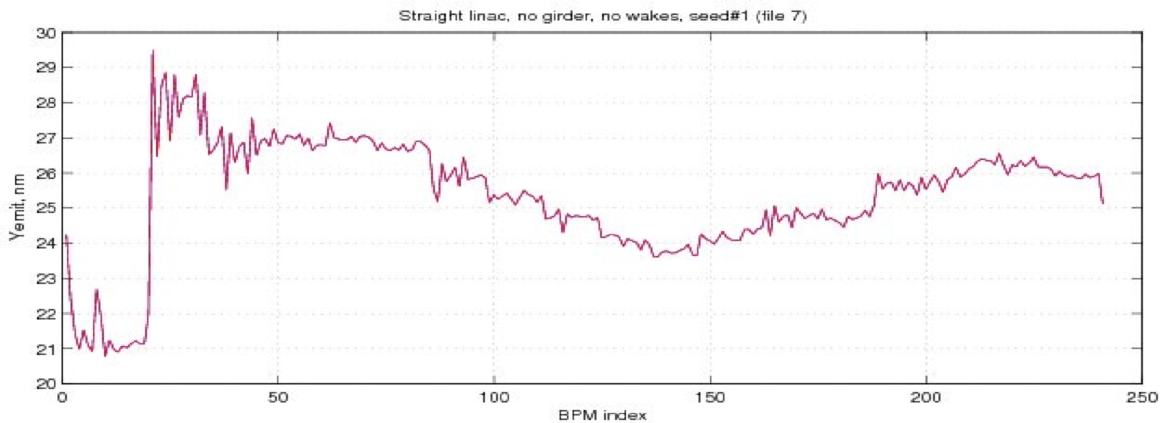


A browser to study the lattice and its hierarchical organization (full support for XSIF format)



User-friendly optics computations and display capabilities. Traditional (uncoupled) or generalized (coupled) lattice functions.

CHEF vs Lucretia Circa 2008



Lucretia

CHEF

Agreement with other codes was generally ok, nevertheless, some differences remained unexplained.

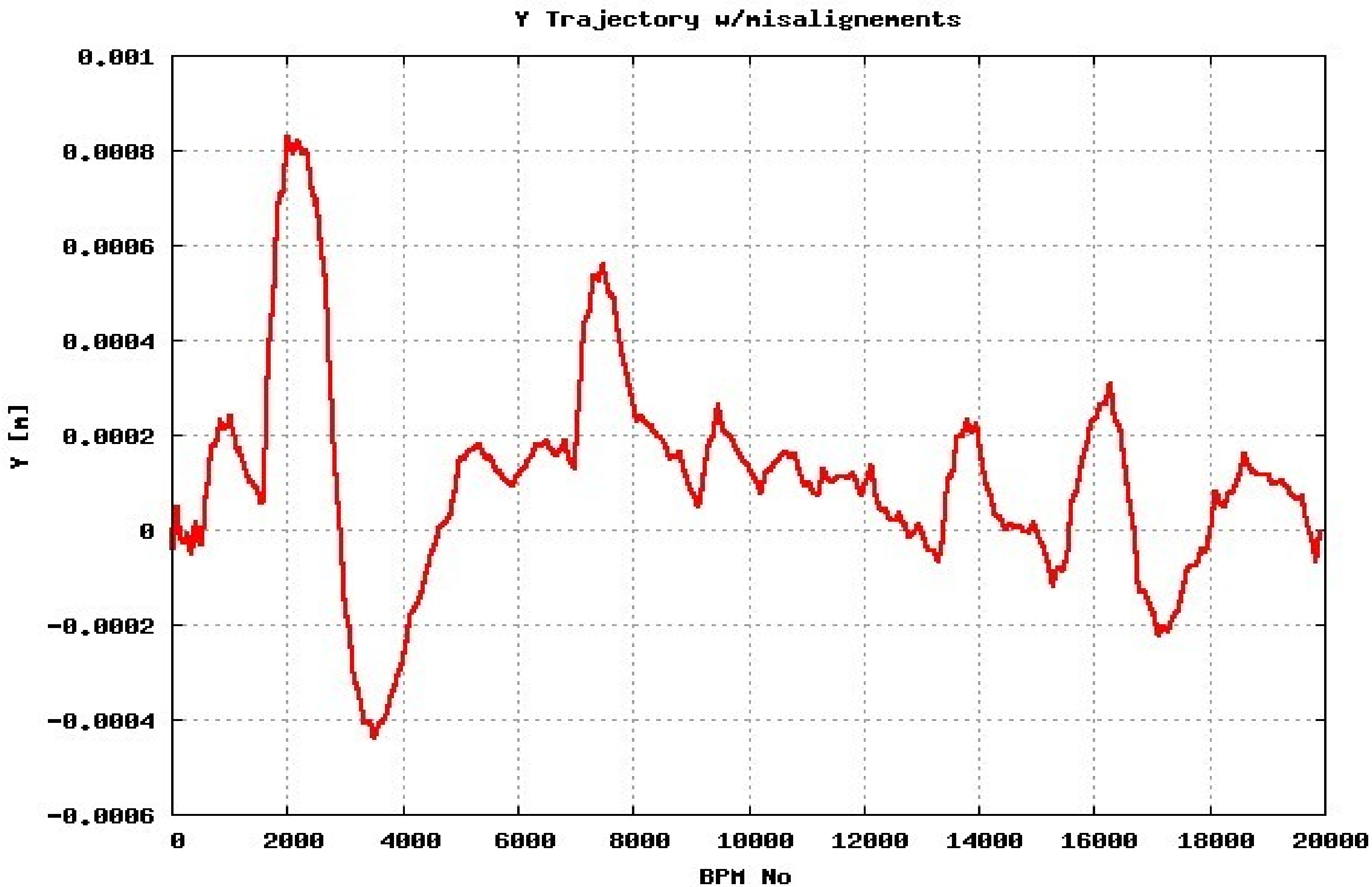
2009: We agree, finally !

What has changed ...

- Cavity tilt angle conventions
- Conversions between canonical and optical variables
- Initial distribution generation (small error in the generated momentum distribution)
- Fixed problem with cavity focusing
- Etc ...

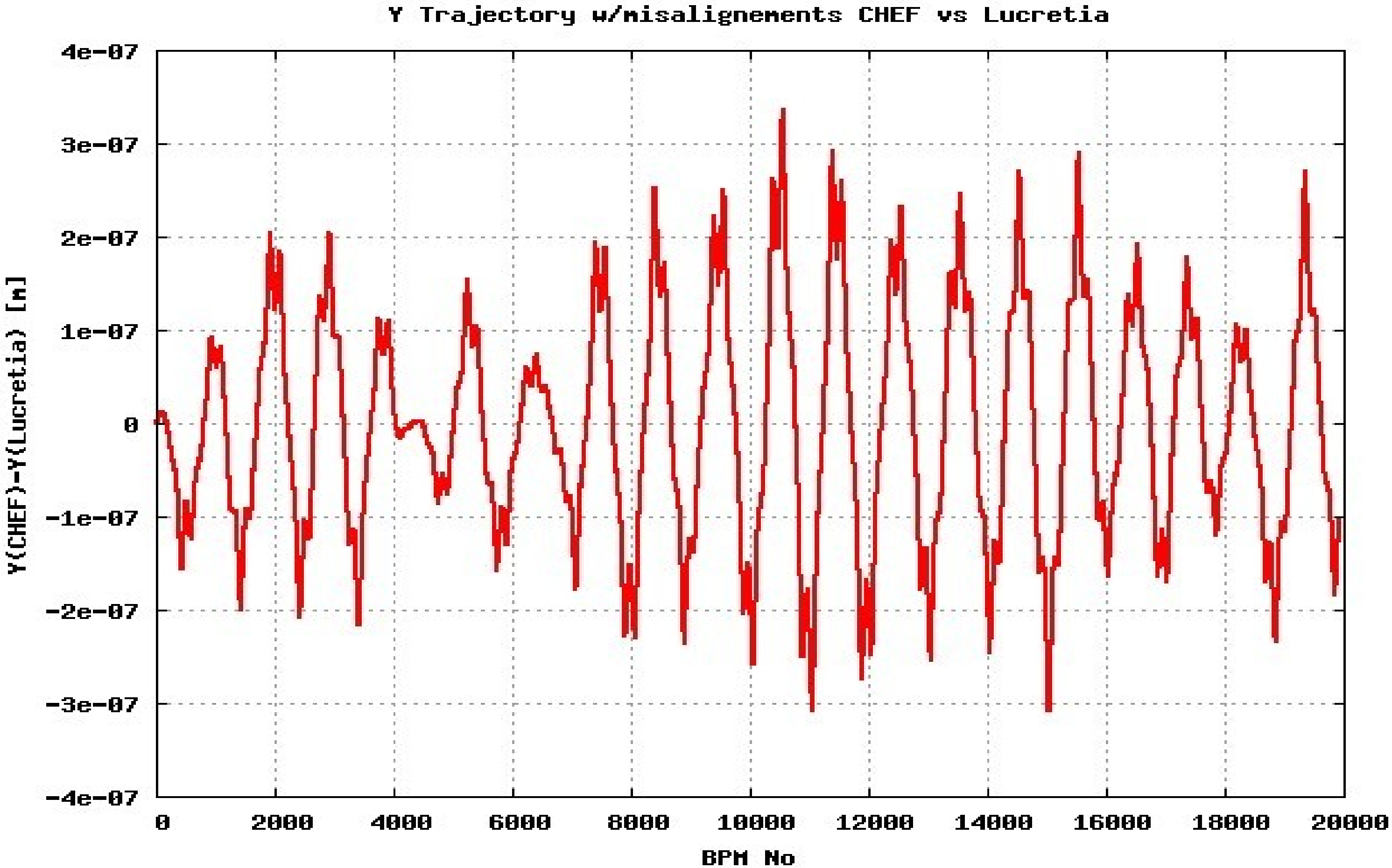
No single dominant factor.

Trajectory in Misaligned Linac



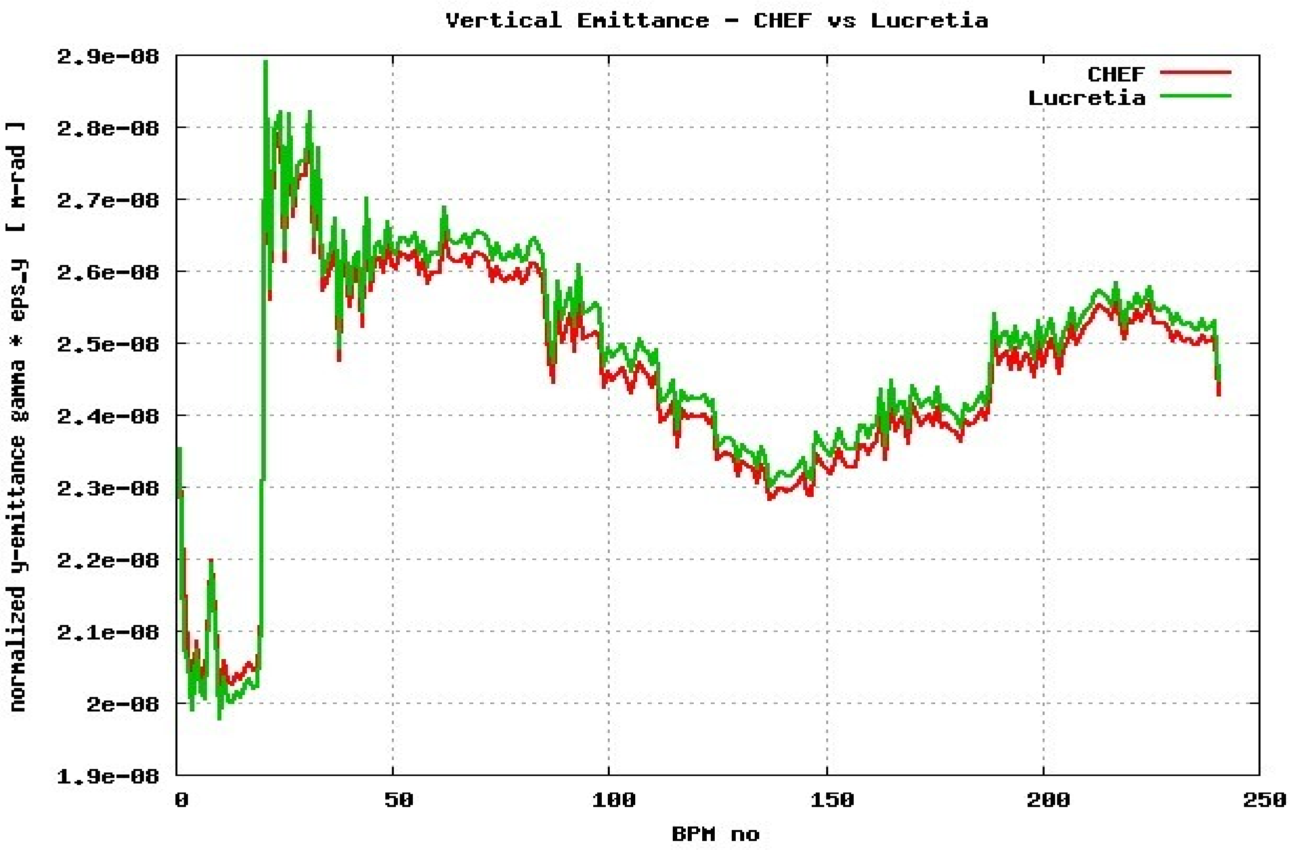
Sample trajectory. Misaligned cavities & quadrupoles

Trajectory Comparison



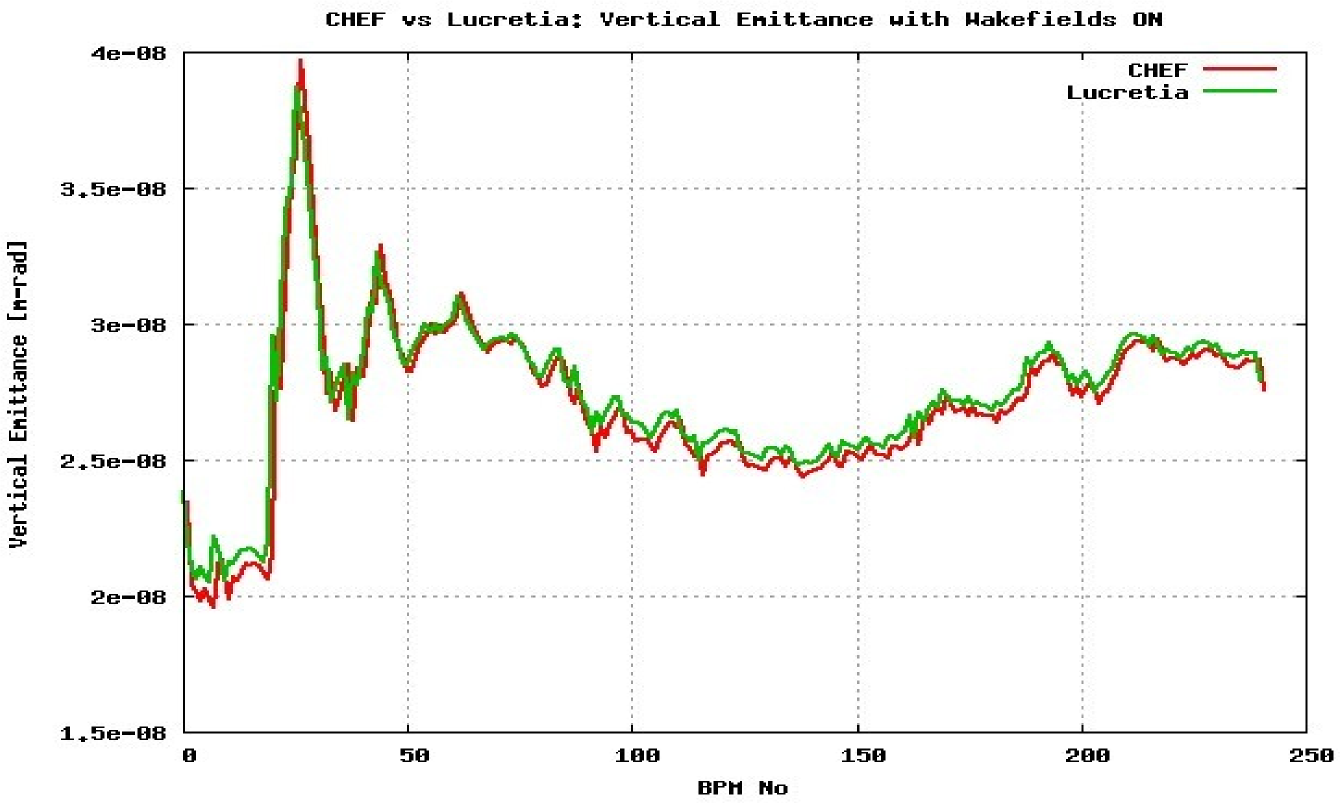
Agreement better than 3.0 E-07 m over 20 km

Ver Emittance after DFS Correction



CHEF vs Lucretia - Wakefields **OFF**

Ver Emittance after DFS Correction

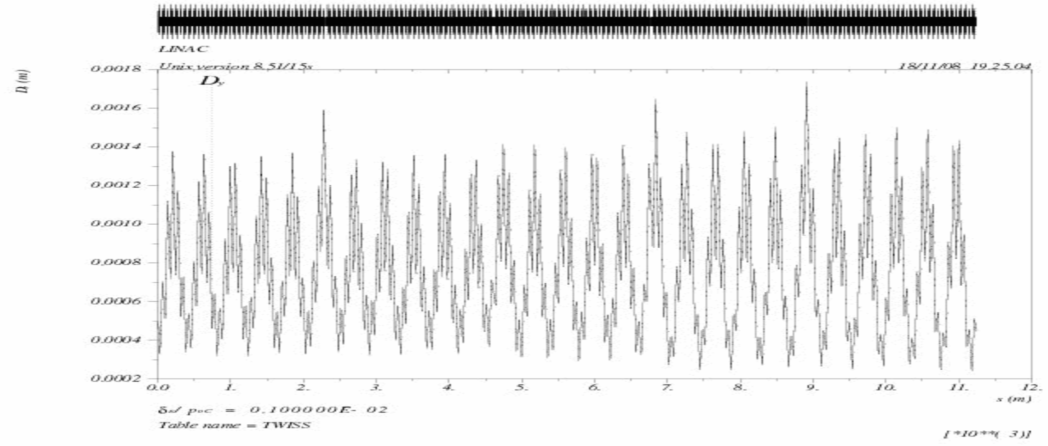


CHEF vs Lucretia: Wakefields ON

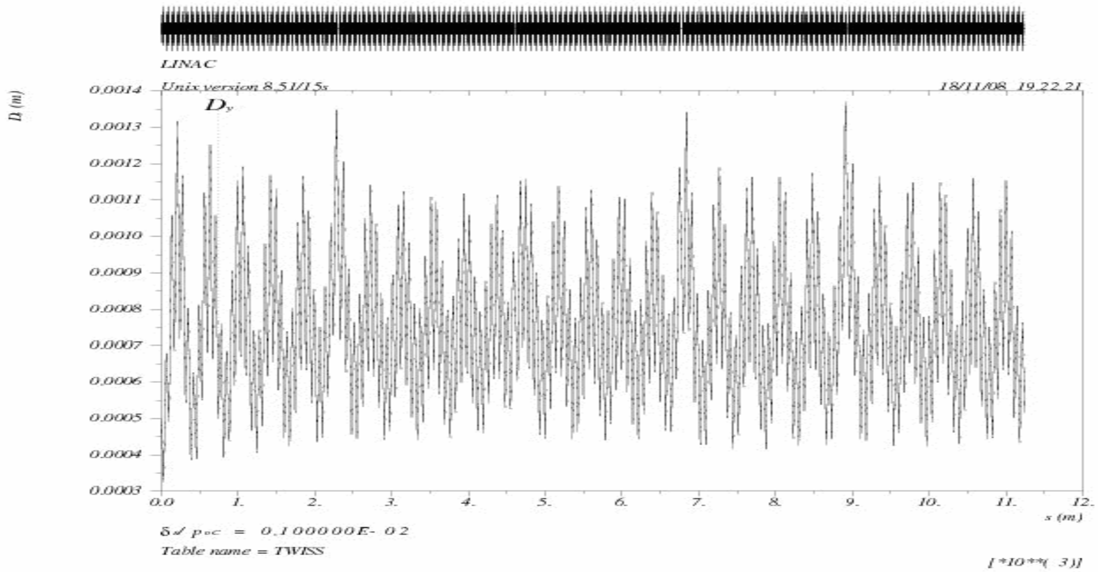
Nominal Dispersion

- We noticed some discrepancies between the nominal dispersion computed by CHEF and other codes.
- The issue is relevant to the extent that optimum corrector settings from the DFS algorithm may be depend on how dispersion is computed.

Dispersion (MADacc)

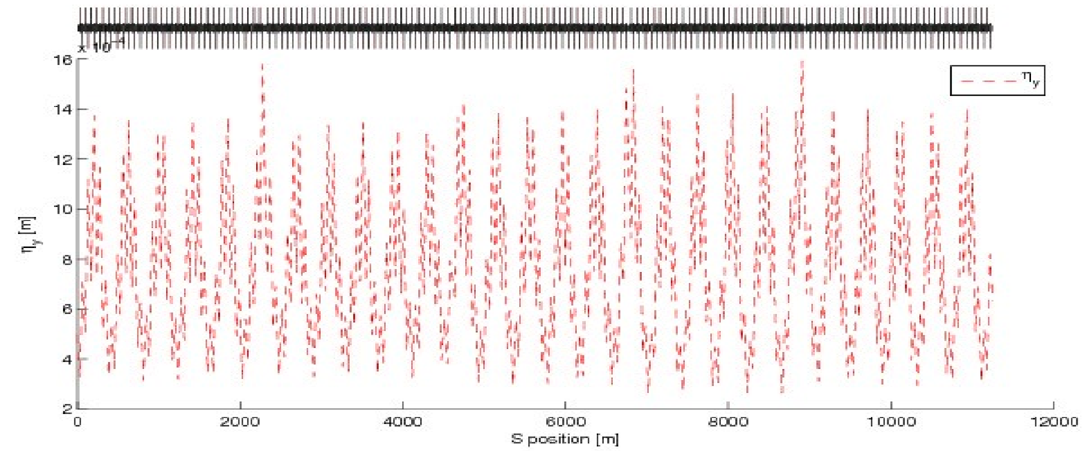


No Acceleration

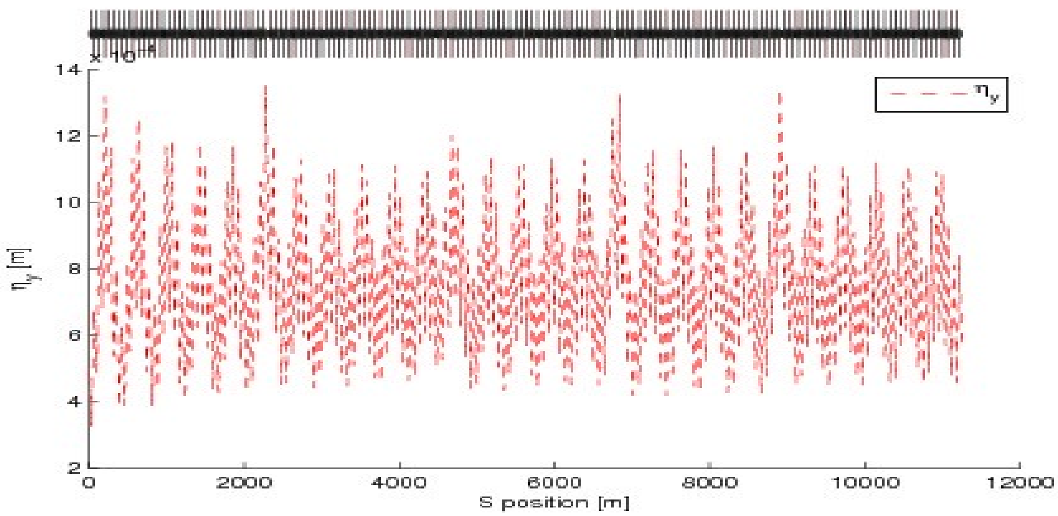


With Acceleration

Dispersion(Lucretia)

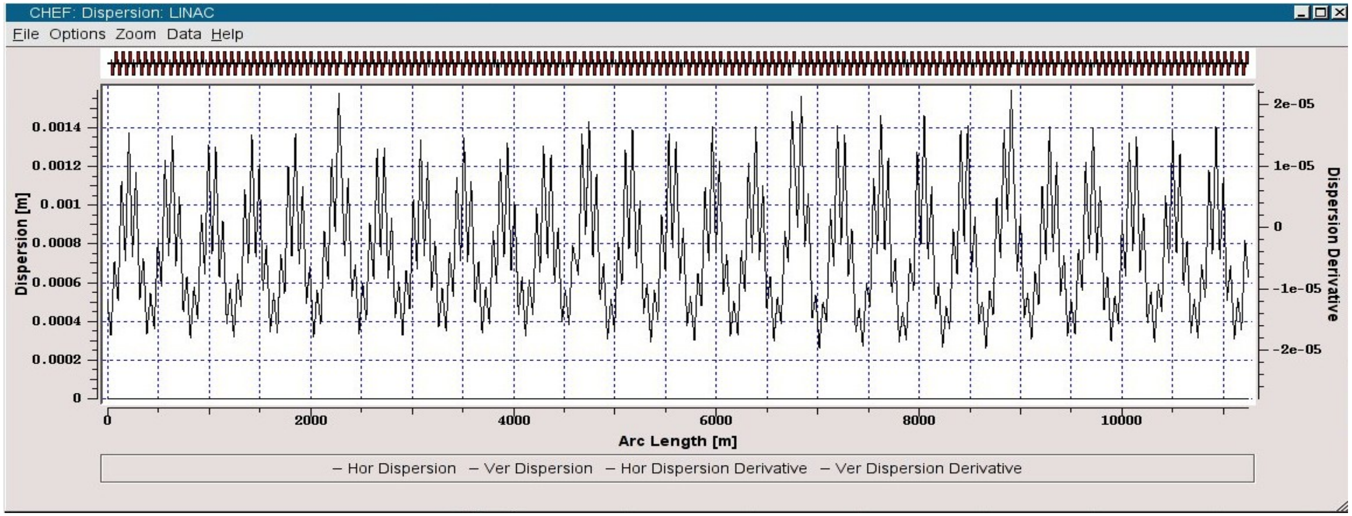


No Acceleration



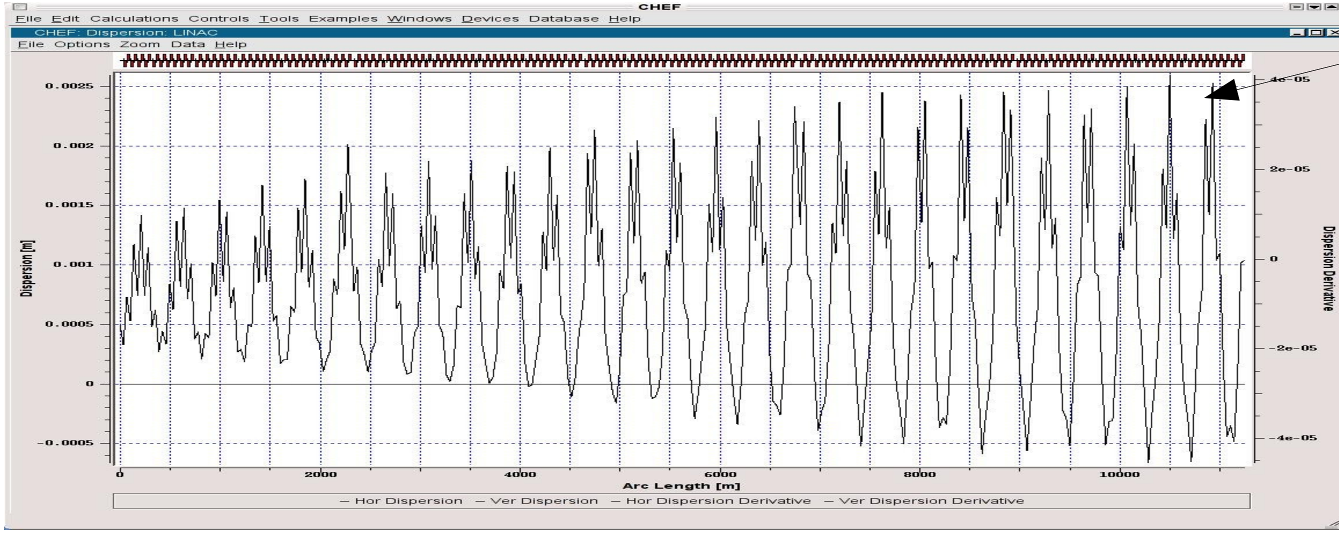
With Acceleration

Dispersion (CHEF)



No Acceleration

25 mm vs 14 mm !



With Acceleration

Remarks

- A curved linac is a bit of an oxymoron ;-)
- In most situations of interest, the nominal dispersion in a linac vanishes (or can be assumed to do so)
- A curved ILC linac is one instance where nominal dispersion is non-zero (albeit small) and matters.
- It is easy to inadvertently compute dispersion under assumptions that are not fully appreciated.

Elementary Dispersion ODE

$$y(s) = y_0(s) + \eta_y(s)\delta(s) \qquad \delta(s) \equiv \delta p(s)/p_0$$

$$\frac{d^2(\eta_y \delta(s))}{ds^2} + K(s)\eta_y \delta(s) = \frac{\delta(s)}{\rho(s)}$$

No acceleration: $\delta = \delta p_0/p_0 = \text{const}$

For a typical HE linac: $\delta(s) = \dfrac{\delta p_0}{p(s)} = \dfrac{\delta p_0}{p_0} \dfrac{1}{(1 + gs/p_0)}$

A Matter of Definition ?

Matrix codes effectively solve

$$\frac{d^2\eta}{ds^2} + K(s)\eta = \frac{1}{\rho(s)} \qquad (1)$$

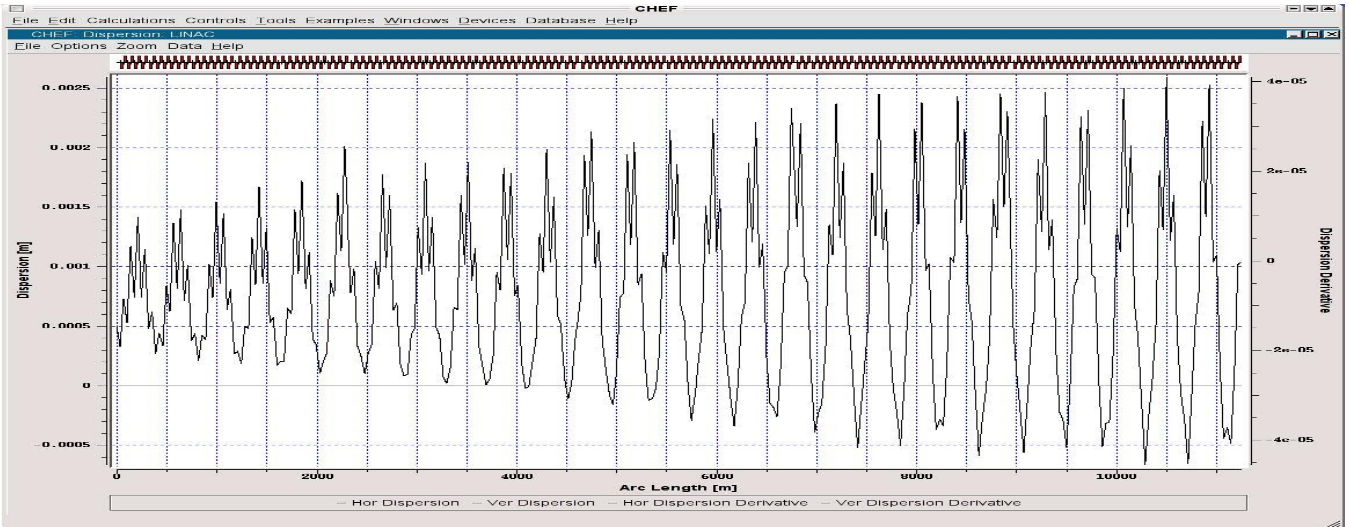
Which is valid only for $\delta = \delta p_0/p_0 = \mathrm{const}$

While it is certainly possible to set the cavity gradients so as to keep dp/p constant, this does not correspond to normal operating conditions.
In general, the dispersion depends on the specific acceleration profile

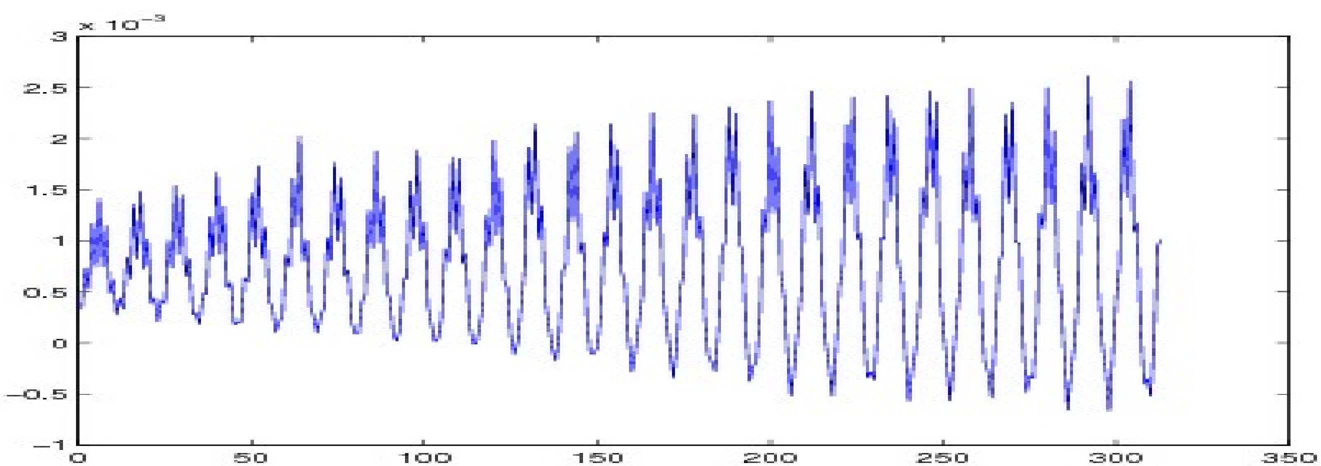
How CHEF computes dispersion

- A reference particle is sent through the linac and the magnet field are scaled so as to make the optical strengths constant for that particle.
- Method 1: A “JetParticle is tracked through the linac. A JetParticle propagates the derivative of y w/r to dp/p to machine precision. The result is scaled by p/p0
- Method2: 2 particles with momenta p and p+dp are tracked through the linac. The result is scaled by p/p0.
- Method3: Same as 2, but this time, the acceleration gradient is modified by a factor 1+dp/p so as to make dp/p constant.

Getting Agreement



CHEF



Lucretia, using a custom script rather than the build-in twiss method.

DFS Algorithm

Minimize

$$\sum w_1(m_j - X_j)^2 + w_2(\Delta m_j - \Delta X_j)^2$$

Where

$m_j, \Delta m_j$ Measured positions and orbit differences

$X_j, \Delta X_j$ Expected (model) positions and orbit differences

The expected difference due to momentum offset is

$$\Delta X_j = \eta(s_j)\delta p/p(s_i)$$

How is η determined ?

Conclusions and Outlook

- Agreement between CHEF and Lucretia is now extremely good; the puzzling small discrepancies observed in recent years are fully understood.
- Other codes like Lucretia and Placet are in routine use at FNAL. In-depth expertise with > 1 code is a valuable asset.
- The manpower dedicated to CHEF is limited; nevertheless, development continues.
- In the course of our studies, we observed some disagreement between the nominal dispersion computed by CHEF and that computed by other codes. The disagreement is understood and results from a different interpretation of 'dispersion'.
- “dispersion” in the context of a linac must be interpreted carefully. In particular, computation of the expected orbit difference induced by a momentum offset in the DFS algorithm may be sensitive to this interpretation.